RETHINKING TARGETED DATA SELECTION FOR INSTRUCTION TUNING: BALANCING TARGET SPECIALIZATION AND CROSS-TASK UTILITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Instruction data selection aims to identify a compact subset of training examples that improves downstream performance under a fixed data budget. Existing target-specific methods primarily optimize relevance to a designated target task, but improvements on the target may coincide with cross-task degradation, while task-balanced methods typically treat multiple tasks symmetrically without explicitly prioritizing the target. We therefore rethink targeted data selection for instruction tuning as an asymmetric objective: improving target specialization while preserving cross-task utility. To address this objective, we propose STAR (Task-**S**hared and **T**arget-Specific **A**ctivation-based **R**anking), which uses cross-task FFN activation statistics from a frozen LLM to construct two complementary neuron sets. Task-Shared neurons exhibit consistently high and relatively balanced activation across reference tasks, while Target-Specific neurons exhibit stronger activation for the designated target than for the remaining reference tasks. These neuron sets define separate activation subspaces, where STAR measures candidate alignment with all reference-task prototypes and the target prototype, respectively, and combines the two scores for selection. We evaluate STAR across five target tasks and four OOD benchmarks under selection budgets of 5% and 1%. Under the 5% budget, STAR achieves the highest Target Avg. of 62.80 and Non-Target Avg. of 60.21, along with the best aligned OOD performance and cross-target OOD transfer among target-specific methods. STAR remains effective under the 1% budget and across backbone models. These results demonstrate that STAR improves target specialization while preserving cross-task utility under limited budgets. The code is available at `https://anonymous.4open.science/r/STAR-42E7/`.

1 INTRODUCTION

Instruction tuning enables LLMs to follow human instructions accurately and supports a wide range of downstream applications (Ouyang et al., 2022; Wei et al., 2022a; Chung et al., 2024). However, modern instruction datasets are often large, heterogeneous, and highly redundant (Xia et al., 2024; Ivison et al., 2025). Training on the complete dataset incurs substantial computational cost and does not necessarily yield optimal downstream performance (Zhou et al., 2023; Chen et al., 2024; Xia et al., 2024). Consequently, selecting a compact yet effective subset from a general-purpose instruction pool has emerged as a central problem in efficient and targeted instruction tuning (Ivison et al., 2023; Xia et al., 2024; Liu et al., 2024; Ivison et al., 2025; Ma et al., 2025). In practical deployments, a model is often adapted to a designated target task while remaining responsible for requests associated with other tasks (Lin et al., 2026; Wu et al., 2023; Chen et al., 2026). For instance, a coding assistant optimized for code generation may still be expected to explain technical concepts and answer documentation-related questions. An effective selection strategy should therefore prioritize the designated target while maintaining performance on the remaining tasks. This requirement motivates the central question of our work: *How can instruction data selection improve target specialization while preserving cross-task utility under a fixed data budget?*

Existing methods do not fully capture this practical requirement. Target-specific approaches, illustrated in Figure 1(a), estimate the relevance of candidate examples to a target task using signals such as gradient influence (Xia et al., 2024; Min et al., 2026), representation alignment (Ivison et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

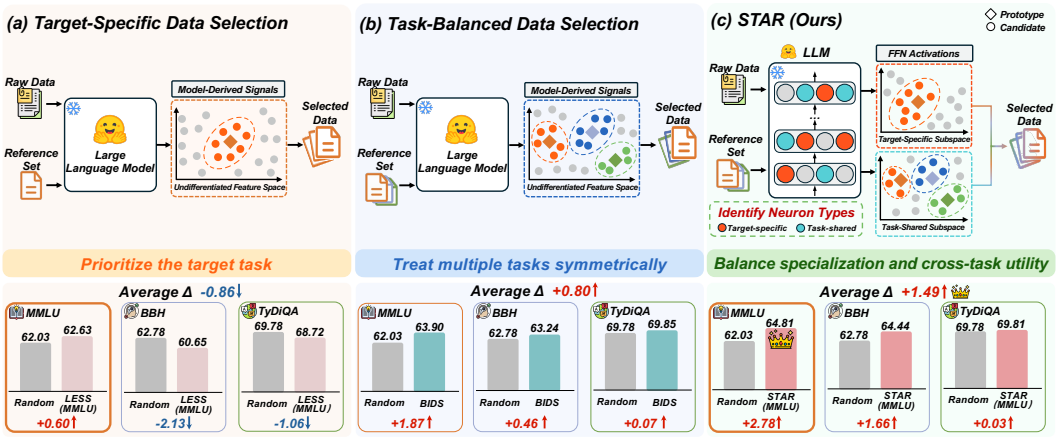


Figure 1: **Comparison of instruction data selection paradigms**. Target-specific selection prioritizes a designated target task, whereas task-balanced selection treats multiple tasks symmetrically. The first two paradigms are schematically represented in a single feature space. STAR instead constructs separate Target-Specific and Task-Shared activation subspaces and combines the resulting signals to balance target specialization and cross-task utility. The bottom panels report performance changes relative to random selection when MMLU serves as the designated target, evaluated on MMLU, BBH, and TyDiQA; Average Δ denotes the mean change across the three benchmarks.

2023; Liu et al., 2024), or neuron activations (Ma et al., 2025; Chen et al., 2026). Although these approaches can improve task-specific performance, their selection objectives are primarily centered on a single target and do not explicitly account for utility on other tasks. Improvements on the target task may therefore coincide with cross-task degradation (Lin et al., 2024; Chen et al., 2026). Task-balanced approaches, illustrated in Figure 1(b), instead emphasize data quality, diversity, or balanced learning across multiple tasks. For example, BIDS treats multiple tasks symmetrically to promote balanced capability acquisition (Dai et al., 2025), but this formulation does not explicitly prioritize a designated target task. Taken together, these limitations motivate us to rethink targeted data selection for instruction tuning as an asymmetric objective, with one selection signal capturing relevance to the designated target and the other capturing cross-task utility.

A key challenge is that when both signals are computed within a single, undifferentiated feature space, task-shared and target-specific patterns remain mixed within a single representation, making it difficult to capture either one separately. This motivates structuring the feature space before candidate scoring. FFN neurons exhibit both cross-task sharing and task selectivity (Song et al., 2024; Leng & Xiong, 2025; Wang et al., 2025b; Tan et al., 2024). Shared neuronal structure is associated with cross-task generalization (Leng & Xiong, 2025; Wang et al., 2025b), and FFN layers encode transferable task-agnostic knowledge (Liu et al., 2026). Motivated by these findings, we introduce two complementary activation criteria over the reference tasks to construct distinct **Task-Shared** and **Target-Specific** neuron sets from a frozen LLM. Task-Shared neurons are selected for consistently high and relatively balanced activation across the reference tasks, whereas Target-Specific neurons are selected for stronger activation on the designated target than on the remaining reference tasks.

Building on these neuron sets, we propose **STAR** (Task-**S**hared and **T**arget-Specific **A**ctivation-based **R**anking), which scores candidate examples in the Task-Shared and Target-Specific activation subspaces defined by the selected neurons. In the Task-Shared subspace, STAR evaluates how strongly and consistently each candidate example aligns with the prototypes of all reference tasks. In the Target-Specific subspace, it measures the alignment between each candidate example and the prototype of the designated target. The two signals are then combined to balance cross-task utility and target relevance.

On Llama-3.1-8B, under the 5% selection budget, STAR achieves the highest Target Avg. of **62.80** across five designated targets, outperforming the SOTA approach by **1.22** points. It also achieves the highest Non-Target Avg. of **60.21**, indicating that the improvement in target specialization is accompanied by strong cross-task utility. Beyond the reference tasks, STAR achieves the best aligned OOD performance and the strongest cross-target OOD transfer among target-specific methods. The

gains remain consistent under the more restrictive 1% selection budget and across different backbone models.

Our main contributions are summarized as follows:

- We revisit the objective of targeted data selection for instruction tuning and formulate a target-prioritized asymmetric setting that aims to improve a designated target task while preserving cross-task utility under a fixed data budget.
- We propose STAR, which constructs Task-Shared and Target-Specific neuron activation subspaces and derives complementary selection signals for candidate instruction examples.
- We conduct systematic evaluations across multiple target tasks, OOD benchmarks, selection budgets, and different backbone models. STAR consistently improves target specialization while maintaining strong cross-task and OOD utility.

2 Related Work

Target-Specific Data Selection for Instruction Tuning. Target-specific data selection seeks training examples that are most useful for a designated downstream task. Influence-based methods estimate target relevance from optimization signals, including optimizer-aware gradient similarity in LESS (Xia et al., 2024), task-specific gradient subspaces in GIST (Min et al., 2026), and learned selection for non-differentiable target metrics in NICE (Wang et al., 2025a). Representation-based methods instead align candidate and target data in a feature space, as in cross-task nearest-neighbor selection and TSDS (Ivison et al., 2023; Liu et al., 2024). Recent work further exploits model-internal activations: MONA (Ma et al., 2025) uses monosemantic activation representations, while NAIT (Chen et al., 2026) selects data from neuron activation patterns. These methods primarily optimize relevance to the designated target, whereas STAR additionally accounts for utility across the remaining reference tasks, since deployed models often remain responsible for diverse requests even after being specialized for a particular application.

Multi-Task and Balanced Data Selection. Related work also considers interactions and utility across multiple tasks or capabilities. Skill-It (Chen et al., 2023) models dependencies among skills to adapt training mixtures, and multi-task instruction tuning studies (Mueller et al., 2024) show that transfer across tasks can materially affect downstream performance. BIDS (Dai et al., 2025) explicitly promotes balanced capability learning by normalizing influence across tasks and prioritizing underrepresented capabilities. Large-scale studies (Ivison et al., 2025; Xia et al., 2025; Nayak et al., 2026) further show that random selection remains a strong baseline and that selection gains depend on data scale and evaluation settings. These approaches motivate preserving broad utility, but generally treat the considered tasks symmetrically. STAR instead targets an asymmetric objective that prioritizes one task while retaining utility across the remaining reference tasks.

Task-Specificity and Cross-Task Sharing in LLM Neurons. Studies of LLM internals identify neurons associated with particular skills, tasks, and languages (Wang et al., 2022; Song et al., 2024; Tang et al., 2024). Other work reveals structured sharing and specialization across tasks and languages (Wang et al., 2025b; Leng & Xiong, 2025; Tan et al., 2024), while transferable task-agnostic knowledge has also been identified within FFN layers (Liu et al., 2026). These findings motivate STAR to derive separate Task-Shared and Target-Specific neuron sets from cross-task activation statistics and use them to construct complementary activation subspaces for data selection.

3 Method

3.1 Problem Formulation and Overview

Let $\mathcal{D} = \{x_i\}_{i=1}^{N}$ denote a general instruction data pool, and let $\mathcal{R} = \{\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_M\}$ denote the reference sets for M tasks, indexed by $m \in \{1, \ldots, M\}$. We designate one task $t \in \{1, \ldots, M\}$ as the target task. Given a data budget B, our objective is to select $\mathcal{D}_t^* \subset \mathcal{D}$ with $|\mathcal{D}_t^*| = B$, such that fine-tuning on $\mathcal{D}_t^*$ improves performance on the designated target while preserving utility on the remaining reference tasks.

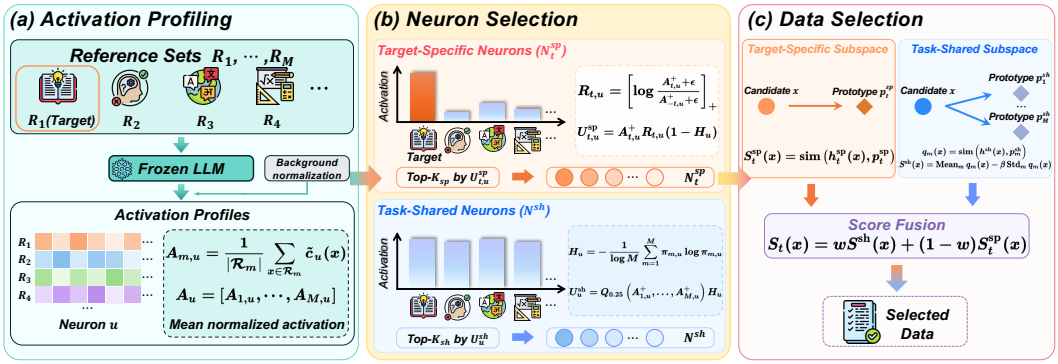


Figure 2: **Overview of STAR**. (a) Cross-task activation profiling from a frozen LLM. (b) Identification of Target-Specific and Task-Shared neurons. (c) Candidate scoring and subset selection in the two activation subspaces.

We use $u = (l, j)$ to index neuron j in FFN layer l. The Task-Shared and Target-Specific neuron sets are denoted by $\mathcal{N}^{\text{sh}}$ and $\mathcal{N}_t^{\text{sp}}$, with corresponding representations $h^{\text{sh}}(x)$ and $h_t^{\text{sp}}(x)$. Their task prototypes are denoted by p_m^{sh} and p_t^{sp}, respectively, and the resulting candidate scores by $S^{\text{sh}}(x)$ and $S_t^{\text{sp}}(x)$. Figure 2 provides an overview of STAR. As shown in Figure 2(a), we first construct normalized cross-task FFN activation profiles. Figure 2(b) then identifies complementary Task-Shared and Target-Specific neuron sets from these profiles. Finally, Figure 2(c) uses the resulting activation subspaces and task prototypes to score candidate examples and select candidate examples. Sections 3.2–3.5 describe these components in detail.

3.2 Normalized FFN Activations

As illustrated in Figure 2(a), we construct normalized cross-task activation profiles for FFN neurons. Specifically, we extract FFN activations from a frozen LLM and normalize them using a common background distribution, enabling consistent comparison of neuron responses across reference tasks.

Following the SwiGLU formulation (Shazeer, 2020; Grattafiori et al., 2024), the gated activation of neuron u at token position s of example x is $a_{u,s}(x) = \text{SiLU}(g_{u,s}(x))\, v_{u,s}(x)$, where $g_{u,s}(x)$ and $v_{u,s}(x)$ denote the outputs of the gate and up projections, respectively. To account for differences in the scale of the down projections, we define the token-level activation score as $c_{u,s}(x) = \log\left(1 + |a_{u,s}(x)|\, \left\|W_u^{\text{dn}}\right\|_2\right)$. We then average the score over token positions:

$$c_u(x) = \frac{1}{|\mathcal{T}(x)|} \sum_{s \in \mathcal{T}(x)} c_{u,s}(x), \tag{1}$$

where $\mathcal{T}(x)$ denotes the token positions used for activation statistics.

Because neurons can have substantially different baseline activation scales, direct comparison of raw scores may favor neurons that are frequently activated by general instruction data. We therefore sample a background set $\mathcal{D}_{\text{bg}}$ from the candidate pool and convert each neuron score into a z-score using its background mean and standard deviation. For reference task m, the average background-normalized activation of neuron u is

$$A_{m,u} = \frac{1}{|\mathcal{R}_m|} \sum_{x \in \mathcal{R}_m} \tilde{c}_u(x), \tag{2}$$

where $\tilde{c}_u(x)$ denotes the background-normalized activation. The cross-task activation profile of neuron u is

$$\mathbf{A}_u = [A_{1,u}, A_{2,u}, \ldots, A_{M,u}]. \tag{3}$$

The resulting profile $\mathbf{A}_u$ summarizes the response of neuron u across the reference tasks and serves as the basis for identifying Task-Shared and Target-Specific neurons.

3.3 Task-Shared and Target-Specific Neuron Selection

In this section, we use the cross-task activation profiles to identify neurons that provide complementary signals for target relevance and cross-task utility. Prior studies show that FFN neurons exhibit both cross-task sharing and task selectivity (Song et al., 2024; Leng & Xiong, 2025; Wang et al., 2025b; Tan et al., 2024). Accordingly, as illustrated in Figure 2(b), STAR selects Task-Shared and Target-Specific neurons using two distinct activation criteria.

Task-Shared Neurons. Task-Shared neurons should exhibit consistently high activation across multiple reference tasks without concentrating on only a few tasks. Because normalized activations can be negative, we first retain their positive components as $A^+_{m,u} = \max(A_{m,u}, 0)$. We normalize the positive activations across tasks and compute the corresponding normalized entropy:

$$\pi_{m,u} = \frac{A^+_{m,u} + \epsilon}{\sum_{r=1}^{M}(A^+_{r,u} + \epsilon)}, \qquad H_u = -\frac{1}{\log M}\sum_{m=1}^{M} \pi_{m,u} \log \pi_{m,u}. \tag{4}$$

We define the Task-Shared score as $U^{\mathrm{sh}}_u = Q_{0.25}\left(A^+_{1,u}, \ldots, A^+_{M,u}\right) H_u$, where $Q_{0.25}$ denotes the lower quartile. This score favors neurons with consistently high positive activation and a balanced activation distribution across tasks.

Target-Specific Neurons. Target-Specific neurons should exhibit high positive activation on the designated target and a clear activation advantage over the remaining reference tasks. For target task t, the average positive activation over the non-target tasks is

$$\overline{A}^+_{-t,u} = \frac{1}{M-1}\sum_{m \neq t} A^+_{m,u}. \tag{5}$$

We quantify the relative target advantage using the positive log activation ratio:

$$R_{t,u} = \left[\log \frac{A^+_{t,u} + \epsilon}{\overline{A}^+_{-t,u} + \epsilon}\right]_+, \tag{6}$$

where $[z]_+ = \max(z, 0)$. We define the Target-Specific score as $U^{\mathrm{sp}}_{t,u} = A^+_{t,u} R_{t,u}(1 - H_u)$. This score favors neurons with strong positive target activation, a large relative advantage over the non-target tasks, and a concentrated cross-task activation distribution.

We select the top K^{sh} and K^{sp} neurons according to the two scores. The resulting neuron sets define two complementary activation subspaces for representing candidate and reference examples. Appendices D.1 and D.2 visualize the layer distribution and activation patterns of the selected neurons.

3.4 Activation Subspaces and Task Prototypes

In this section, we construct example representations and task prototypes in the Task-Shared and Target-Specific activation subspaces, as illustrated in Figure 2(c). For an example x, the corresponding representations are

$$\phi^{\mathrm{sh}}(x) = \left[\widetilde{c}_u(x)\right]_{u \in \mathcal{N}^{\mathrm{sh}}}, \qquad \phi^{\mathrm{sp}}_t(x) = \left[\widetilde{c}_u(x)\right]_{u \in \mathcal{N}^{\mathrm{sp}}_t}. \tag{7}$$

Both representations are L2-normalized and denoted by $h^{\mathrm{sh}}(x)$ and $h^{\mathrm{sp}}_t(x)$.

Let $\mathrm{norm}(v) = v/(\|v\|_2 + \epsilon)$. The Task-Shared prototype of reference task m and the Target-Specific prototype of the designated target t are defined as

$$p^{\mathrm{sh}}_m = \mathrm{norm}\left(\frac{1}{|\mathcal{R}_m|}\sum_{x \in \mathcal{R}_m} h^{\mathrm{sh}}(x)\right), \qquad p^{\mathrm{sp}}_t = \mathrm{norm}\left(\frac{1}{|\mathcal{R}_t|}\sum_{x \in \mathcal{R}_t} h^{\mathrm{sp}}_t(x)\right). \tag{8}$$

The resulting prototypes serve as reference anchors for measuring candidate alignment in the two activation subspaces.

3.5 Data Scoring and Selection

As shown in Figure 2(c), we score each candidate example in the two activation subspaces and combine the resulting signals for subset selection. For candidate example x, its similarity to the Task-Shared prototype of task m is $q_m(x) = \mathrm{sim}(h^{\mathrm{sh}}(x), p_m^{\mathrm{sh}})$, where $\mathrm{sim}(\cdot,\cdot)$ denotes cosine similarity. The Task-Shared score is

$$S^{\mathrm{sh}}(x) = \mathrm{Mean}_m\, q_m(x) - \beta\, \mathrm{Std}_m\, q_m(x), \tag{9}$$

where β controls the penalty for uneven alignment across tasks. The mean term rewards high overall alignment, whereas the standard-deviation term penalizes candidates aligned with only a few tasks.

The Target-Specific score is $S_t^{\mathrm{sp}}(x) = \mathrm{sim}(h_t^{\mathrm{sp}}(x), p_t^{\mathrm{sp}})$. We convert the two scores into percentile ranks, denoted by $\widehat{S}^{\mathrm{sh}}(x)$ and $\widehat{S}_t^{\mathrm{sp}}(x)$, and combine them as

$$S_t(x) = w\widehat{S}^{\mathrm{sh}}(x) + (1-w)\widehat{S}_t^{\mathrm{sp}}(x). \tag{10}$$

The weight $w \in [0,1]$ controls the trade-off between cross-task utility and target relevance.

Selecting examples solely according to their individual scores can produce a redundant subset because pointwise ranking does not explicitly account for diversity among the selected examples. Prior data selection methods have therefore incorporated mechanisms such as clustering, balanced sampling, diversity regularization, and redundancy-aware selection to improve subset diversity (Ge et al., 2024; Yang et al., 2024; Liu et al., 2024; Zhang et al., 2026). For each candidate example x, we define its joint representation for clustering as $z_t(x) = [h^{\mathrm{sh}}(x); h_t^{\mathrm{sp}}(x)]$, where $[\cdot;\cdot]$ denotes vector concatenation. Following this general practice, STAR clusters candidate examples according to their joint representations and prioritizes high-scoring examples within each cluster. This cluster-aware procedure preserves the ranking signal while promoting diversity in the resulting subset.

4 Experiments

4.1 Experimental Setup

Training Data and Evaluation Tasks. Following prior work on targeted data selection for instruction tuning (Xia et al., 2024; Ma et al., 2025), we use the same approximately 270K-example instruction pool, composed of FLAN v2 (Longpre et al., 2023), Chain-of-Thought (Wei et al., 2022b), Dolly (Conover et al., 2023), and OpenAssistant (Köpf et al., 2023). All selection methods operate on the same candidate pool.

We consider five reference tasks spanning distinct capabilities: MMLU (Hendrycks et al., 2021a) for factual knowledge and multi-domain understanding, BBH (Suzgun et al., 2023) for complex reasoning, TyDiQA (Clark et al., 2020) for multilingual question answering, GSM8K (Cobbe et al., 2021) for mathematical reasoning, and MBPP (Austin et al., 2021) for code generation. Each task serves as the designated target in turn, while the remaining four serve as non-target reference tasks, yielding five selection settings, one for each target task.

Following the OOD evaluation protocol of NAIT (Chen et al., 2026), we additionally evaluate MMLU-Pro (Wang et al., 2024), XCOPA (Ponti et al., 2020), MATH (Hendrycks et al., 2021b), and HumanEval (Chen et al., 2021) as OOD benchmarks, all excluded from data selection. Aligned OOD Avg. averages four predefined capability pairings: MMLU–MMLU-Pro, TyDiQA–XCOPA, GSM8K–MATH, and MBPP–HumanEval; for target-specific methods, each OOD benchmark is evaluated using the model trained for its paired target task. Cross-Target OOD Avg. averages each OOD benchmark across the five models trained for different target tasks and then across the four benchmarks. Methods without target-specific selection are evaluated directly on the OOD benchmarks, while Target Avg. and Non-Target Avg. summarize performance on the designated and remaining reference tasks, respectively.

Models and Training. We use Llama-3.1-8B (Grattafiori et al., 2024) as the primary backbone and additionally evaluate Qwen2.5-7B (Qwen et al., 2025) to assess consistency across backbone models. For STAR, we set $K^{\mathrm{sh}} = K^{\mathrm{sp}} = 4096$ and use $w = 0.5$ to combine the two selection signals. Before subset construction, we partition the candidate examples into 100 clusters to reduce redundancy among high-scoring examples.

Table 1: **Performance across five reference tasks and four OOD benchmarks under different selection budgets.** For methods that select a separate subset for each target task, the OOD columns use the predefined MMLU–MMLU-Pro, TyDiQA–XCOPA, GSM8K–MATH, and MBPP–HumanEval pairings. Methods without target-specific selection are evaluated directly on the OOD benchmarks. Target Avg. and Aligned OOD Avg. average the five target-task results and four OOD results, respectively.

Method	Factual Knowledge		Multilingual		Reasoning	Math		Code		Average	
	MMLU	MMLU-Pro	TyDiQA	XCOPA	BBH	GSM8K	MATH	MBPP	H-Eval	Target Avg.	Aligned OOD Avg.
Base	65.64	35.94	69.73	62.47	64.72	52.01	19.20	48.00	37.20	60.02	38.70
Full	61.37	34.06	70.33	62.35	63.70	63.00	16.78	47.60	40.85	61.20	38.51
5% Selection Budget											
Random	62.03	36.82	69.78	62.35	62.78	54.89	18.24	46.60	43.29	59.22	40.17
LESS	62.63	34.45	71.71	62.78	63.89	58.00	19.24	47.00	39.63	60.65	39.03
MONA	63.58	34.64	**72.26**	62.93	64.26	60.42	17.42	47.40	44.51	61.58	39.88
NAIT	64.24	**37.30**	71.45	62.35	64.17	55.50	18.76	48.20	40.24	60.71	39.66
BIDS	63.90	35.46	69.85	62.53	63.24	56.79	17.70	45.80	**45.73**	59.92	40.36
STAR	**64.81**	37.28	72.01	**63.25**	**65.65**	**62.93**	**19.94**	**48.60**	43.90	**62.80**	**41.09**
1% Selection Budget											
Random	65.33	37.12	69.08	62.71	67.78	57.54	20.70	48.00	42.07	61.55	40.65
LESS	63.23	35.26	70.26	63.25	67.87	58.61	20.50	47.60	40.85	61.51	39.97
MONA	64.65	35.51	68.12	62.40	65.74	62.55	18.48	47.40	**44.51**	61.69	40.23
NAIT	65.41	36.92	67.41	62.33	65.65	59.14	**21.08**	48.00	42.68	61.12	40.75
BIDS	63.94	37.15	**71.04**	62.18	**68.06**	55.72	20.64	48.40	43.29	61.43	40.82
STAR	**66.02**	**38.57**	70.69	**63.44**	66.67	**63.31**	20.50	**49.40**	**44.51**	**63.22**	**41.76**

We evaluate selection budgets of 5% and 1% of the candidate pool. All models are fine-tuned for two epochs using the same training setup, implemented with LLaMA-Factory (Zheng et al., 2024). Additional training configurations, implementation details, and hyperparameter settings are provided in the Appendix.

Baselines. We compare STAR with representative baselines covering target-specific and task-balanced data selection. Full denotes training on the entire candidate pool, while Random uniformly samples examples under the same selection budget. LESS (Xia et al., 2024) performs target-specific selection by estimating data influence using gradient similarity between candidate and target examples. MONA (Ma et al., 2025) selects task-relevant examples by aligning neuron activation representations with a target prototype. NAIT (Chen et al., 2026) identifies target-relevant training data based on neuron activation patterns. BIDS (Dai et al., 2025) treats multiple capabilities symmetrically to promote balanced learning across tasks. Together, these baselines cover full-data training, random selection, target-specific selection, and task-balanced selection.

4.2 Main Results

Target Specialization. Table 1 reports target-task performance under the 5% and 1% selection budgets. Under the 5% budget, STAR achieves the highest Target Avg. of 62.80, outperforming the strongest selection baseline, MONA, by **1.22** points and full-data training by **1.60** points. Under the more restrictive 1% budget, STAR reaches a Target Avg. of 63.22, exceeding MONA by **1.53** points.

At the task level, under the 5% budget, STAR ranks first among the selection methods on MMLU, BBH, GSM8K, and MBPP, and second on TyDiQA. Under the 1% budget, it achieves the best performance on MMLU, GSM8K, and MBPP while maintaining the highest Target Avg. These results show consistent gains across multiple target tasks.

Cross-Task Utility. Target-task performance alone does not fully characterize the utility of a selected subset. For each target setting, we therefore evaluate the resulting model on the remaining four reference tasks and use their average to measure non-target utility. Non-Target Avg. further averages this quantity across the five target settings. Table 2 reports target-task performance and non-target utility for each setting.

Table 2: **Target specialization and cross-task utility under the 5% selection budget**. For each target setting, ***Target*** reports performance on the designated target task, and ***Non-Target*** reports the average over the other four reference tasks. Random selects a single target-agnostic subset and is shown as a reference, excluded from ranking. Best and second-best selection results in each column are bolded and underlined, respectively.

Method	MMLU		BBH		TyDiQA		GSM8K		MBPP		Avg.	
	Target	*Non-Target*	*Target*	*Non-Target*	*Target*	*Non-Target*	*Target*	*Non-Target*	*Target*	*Non-Target*	*Target*	*Non-Target*
Random	62.03	58.51	62.78	58.33	69.78	56.58	54.89	60.30	46.60	62.37	59.22	59.22
LESS	62.63	54.51	63.89	<u>59.70</u>	71.71	53.00	58.00	60.67	47.00	61.23	60.65	57.82
MONA	63.58	55.32	<u>64.26</u>	57.55	**72.26**	<u>57.35</u>	<u>60.42</u>	58.85	47.40	<u>62.79</u>	<u>61.58</u>	58.37
NAIT	<u>64.24</u>	<u>57.36</u>	64.17	59.01	71.45	57.06	55.50	<u>60.91</u>	<u>48.20</u>	61.42	60.71	<u>59.15</u>
STAR	**64.81**	**58.35**	**65.65**	**60.14**	<u>72.01</u>	**57.76**	**62.93**	**61.20**	**48.60**	**63.60**	**62.80**	**60.21**

Table 3: **Cross-target OOD transfer under the 5% selection budget.** Each OOD result is averaged across the five target settings, with ± reporting the standard deviation across settings; Cross-Target OOD Avg. further averages over the four benchmarks. Random and BIDS perform target-agnostic selection and are shown as references. Best and second-best results among target-specific methods are bolded and underlined, respectively.

Method	MMLU-Pro	XCOPA	MATH	HumanEval	Cross-Target OOD Avg.
Random	36.82	62.35	18.24	43.29	40.17
BIDS	35.46	62.53	17.70	45.73	40.36
LESS	34.87 ±0.98	62.61 ±0.12	15.66 ±1.99	40.61 ±2.90	38.44
MONA	35.76 ±1.00	<u>62.90</u> ±2.29	14.58 ±4.36	<u>41.34</u> ±1.78	38.65
NAIT	<u>35.81</u> ±1.27	62.58 ±0.41	<u>17.10</u> ±1.00	40.73 ±0.46	<u>39.06</u>
STAR	**36.48** ±0.60	**63.18** ±0.41	**17.75** ±1.24	**42.93** ±0.91	**40.08**

Random provides strong cross-task utility, with a Non-Target Avg. of 59.22. STAR further improves this value to 60.21 while increasing Target Avg. from 59.22 to 62.80. These results show that the improvement in target specialization is accompanied by strong utility on the remaining reference tasks, demonstrating a favorable balance between target specialization and cross-task utility.

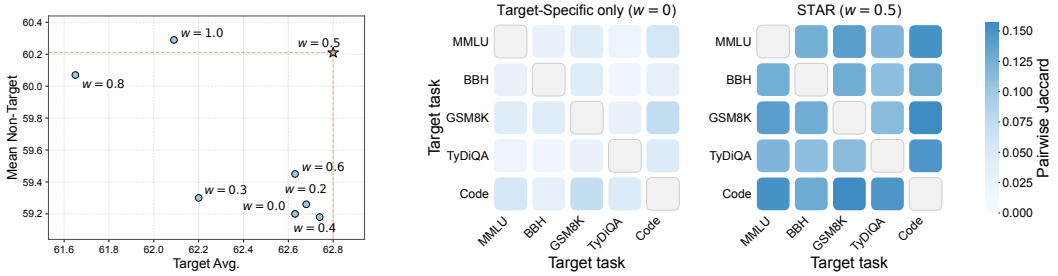
OOD Transfer. We evaluate OOD transfer from two complementary perspectives. Table 1 reports aligned OOD transfer based on the predefined broad-capability pairings. Under the 5% budget, STAR achieves the highest Aligned OOD Avg. of 41.09. The same trend remains under the 1% budget, where STAR reaches 41.76.

Table 3 provides a broader cross-target evaluation by averaging performance on each OOD benchmark across the five models trained for different target tasks. Among the target-specific methods, STAR achieves the highest Cross-Target OOD Avg. of 40.08, followed by NAIT at 39.06. The performance of STAR is also comparable to the target-agnostic Random and BIDS baselines, which obtain 40.17 and 40.36, respectively. Together with the aligned OOD results, these findings show that STAR improves target specialization while preserving strong OOD utility beyond the designated target.

Additional Backbone. We further evaluate STAR on Qwen2.5-7B under the 5% selection budget to assess consistency across backbone models. As shown in Table 4, STAR achieves the highest Target Avg. and Non-Target Avg. among the compared methods, together with the second-highest Cross-Target OOD Avg. Because BIDS uses a single target-agnostic balanced subset, its Target Avg. and Non-Target Avg. are identical, while it achieves the highest Cross-Target OOD Avg. These results indicate that the gains in target specialization and cross-task utility observed on Llama-3.1-8B also extend to Qwen2.5-7B.

Table 4: Results on Qwen2.5-7B under the 5% selection budget. Best and second-best results are bolded and underlined, respectively.

Method	Target Avg.	Non-Target Avg.	Cross-Target OOD Avg.
LESS	68.98	68.41	48.29
NAIT	<u>69.04</u>	68.43	48.71
BIDS	68.97	<u>68.97</u>	**50.01**
STAR	**69.66**	**69.58**	<u>49.31</u>



(a) Sensitivity to the Shared–Specific balance.

(b) Overlap among subsets selected for different target tasks.

Figure 3: **Analysis of the Shared–Specific balance**. (a) Trade-off between Target Avg. and Non-Target Avg. across different values of w. (b) Pairwise overlap among subsets selected for different target tasks under Target-Specific-only selection ($w = 0$) and STAR ($w = 0.5$).

5 Analysis

5.1 Effect of Neuron Selection

We examine whether separating Task-Shared and Target-Specific patterns into dedicated activation subspaces is more effective than computing both signals in a single, undifferentiated FFN space. Table 5 compares STAR with all-neuron and random-neuron variants while keeping the remaining selection pipeline unchanged. STAR achieves the best Target Avg., Non-Target Avg., and Cross-Target OOD Avg. (labeled OOD Avg. in the table) of 62.80, 60.21, and 40.08, respectively. These results show that measuring the two signals in their dedicated activation subspaces is more effective than using a common or randomly selected FFN space. Sensitivity to the neuron number and clustering is reported in Appendices C.1 and C.2.

Table 5: **Effect of FFN neuron selection under the 5% selection budget**.

Neuron Set	Target Avg.	Non-Target Avg.	OOD Avg.
All	61.31	<u>58.95</u>	<u>38.97</u>
Random	<u>61.73</u>	57.90	38.42
STAR-selected	**62.80**	**60.21**	**40.08**

5.2 Shared–Specific Balance

Figure 3(a) illustrates the trade-off between target specialization and cross-task utility across the fusion weight w. Target-Specific-only selection ($w = 0$) achieves a Target Avg. of 62.63 with a Non-Target Avg. of 59.20, whereas Task-Shared-only selection ($w = 1$) improves the Non-Target Avg. to 60.29 with a lower Target Avg. of 62.09. At $w = 0.5$, STAR achieves the highest Target Avg. of 62.80 while maintaining a strong Non-Target Avg. of 60.21, showing that the two signals provide complementary benefits for balancing target specialization and cross-task utility.

Figure 3(b) further compares the overlap among subsets selected for different target tasks. Relative to Target-Specific-only selection, STAR yields higher pairwise overlap across target settings while the selected subsets remain distinct, suggesting that the Task-Shared signal introduces common cross-task structure while preserving target-specific differences. Appendix B.2 further shows that providing MONA and NAIT with the same five reference tasks does not reproduce the performance of STAR, highlighting the importance of how the Task-Shared and Target-Specific signals are constructed and combined.

6 Conclusion

We revisit targeted data selection for instruction tuning to improve target specialization while preserving cross-task utility. We propose STAR, which combines Task-Shared and Target-Specific signals derived from FFN activations of a frozen LLM. Across five target tasks, STAR achieves the highest Target Avg. under both the 5% and 1% budgets while maintaining strong performance on non-target tasks and OOD benchmarks. Further analysis shows that combining the two signals provides a favorable balance between target specialization and cross-task utility.

AI USE STATEMENT

We used generative AI tools to assist with language editing and to support qualitative analysis of the characteristics of selected training examples. The qualitative analysis was conducted only after data selection and model evaluation and did not affect the selection procedure, training process, evaluation metrics, or reported quantitative results. We did not use generative AI tools to generate experimental data or determine numerical results. All AI-assisted analyses and text were reviewed by the authors for technical correctness and consistency with the underlying data. We take responsibility for the final content of this work, including all text, claims, analyses, and results.

ETHICS STATEMENT

This work uses publicly available datasets and open-source models. It does not involve human subjects, private or sensitive data, or other procedures that raise specific ethical concerns. We do not identify any specific ethical risks associated with this work.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the STAR method and experimental setup in the main paper and appendix. The appendix specifies the construction of the reference and background sets, implementation details, training and evaluation configurations, and baseline implementations. We also report additional task-level results, ablation studies, and analyses to support reproducibility. Anonymous code is available at `https://anonymous.4open.science/r/STAR-42E7/`.

REFERENCES

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models, 2021. URL `https://arxiv.org/abs/2108.07732`.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, and Hongxia Jin. Alpagasus: Training a better alpaca with fewer data. In *The Twelfth International Conference on Learning Representations*, 2024.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021. URL `https://arxiv.org/abs/2107.03374`.

Mayee Chen, Nicholas Roberts, Kush Bhatia, Jue Wang, Ce Zhang, Frederic Sala, and Christopher Ré. Skill-it! a data-driven skills framework for understanding and training language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL `https://proceedings.neurips.cc/paper_files/paper/2023/hash/70b8505ac79e3e131756f793cd80eb8d-Abstract-Conference.html`.

Xin Chen, Junchao Wu, Shu Yang, Runzhe Zhan, Zeyu Wu, Min Yang, Shujian Huang, Lidia S. Chao, and Derek F. Wong. Neuron-aware data selection in instruction tuning for large language models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL `https://openreview.net/forum?id=uq6UWRgzMr`.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin

Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024. URL http://jmlr.org/papers/v25/23-0870.html.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470, 2020. doi: 10.1162/tacl_a_00317. URL https://aclanthology.org/2020.tacl-1.30/.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. Free dolly: Introducing the world's first truly open instruction-tuned LLM. Databricks Blog, 2023. URL https://www.databricks.com/blog/2023/04/12/dolly-first-open-commercially-viable-instruction-tuned-llm.

Qirun Dai, Dylan Zhang, Jiaqi W. Ma, and Hao Peng. Improving influence-based instruction tuning data selection for balanced learning of diverse capabilities. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 7079–7102. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-emnlp.373. URL https://aclanthology.org/2025.findings-emnlp.373/.

Yuan Ge, Yilun Liu, Chi Hu, Weibin Meng, Shimin Tao, Xiaofeng Zhao, Hongxia Ma, Li Zhang, Boxing Chen, Hao Yang, Bei Li, Tong Xiao, and Jingbo Zhu. Clustering and ranking: Diversity-preserved instruction selection through expert-aligned quality estimation. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 464–478, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.28. URL https://aclanthology.org/2024.emnlp-main.28/.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In J. Vanschoren and S. Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021b. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/be83ab3ecd0db773eb2dc1b0a17836a1-Paper-round2.pdf.

Hamish Ivison, Noah A. Smith, Hannaneh Hajishirzi, and Pradeep Dasigi. Data-efficient finetuning using cross-task nearest neighbors. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 9036–9061. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-acl.576. URL https://aclanthology.org/2023.findings-acl.576/.

Hamish Ivison, Muru Zhang, Faeze Brahman, Pang Wei Koh, and Pradeep Dasigi. Large-scale data selection for instruction tuning, 2025. URL https://arxiv.org/abs/2503.01807.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, Shahul ES, Sameer Suri, David Glushkov, Arnav Dantuluri, Andrew Maguire, Christoph Schuhmann, Huu Nguyen, and Alexander Mattick. OpenAssistant Conversations: Democratizing large language model alignment. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-2064.

Yongqi Leng and Deyi Xiong. Towards understanding multi-task learning (generalization) of LLMs via detecting and exploring task-specific neurons. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 2969–2987, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics. URL https://aclanthology.org/2025.coling-main.200/.

Jiacheng Lin, Zhongruo Wang, Kun Qian, Tian Wang, Arvind Srinivasan, Hansi Zeng, Ruochen Jiao, Xie Zhou, Jiri Gesi, Dakuo Wang, Yufan Guo, Kai Zhong, Weiqi Zhang, Sujay Sanghavi, Changyou Chen, Hyokun Yun, and Lihong Li. SFT doesn't always hurt general capabilities: Revisiting domain-specific fine-tuning in LLMs. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 46954–46992, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/4e447acb68f57e29234bc0eb19896f11-Paper-Conference.pdf.

Yong Lin, Hangyu Lin, Wei Xiong, Shizhe Diao, Jianmeng Liu, Jipeng Zhang, Rui Pan, Haoxiang Wang, Wenbin Hu, Hanning Zhang, Hanze Dong, Renjie Pi, Han Zhao, Nan Jiang, Heng Ji, Yuan Yao, and Tong Zhang. Mitigating the alignment tax of RLHF. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 580–606, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.35. URL https://aclanthology.org/2024.emnlp-main.35/.

Chang Liu, Boyu Shi, Xu Yang, Qiufeng Wang, and Xin Geng. Inheriting generalizable knowledge from LLMs to diverse vertical tasks. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=m38vyzeoO0.

Zifan Liu, Amin Karbasi, and Theodoros Rekatsinas. TSDS: Data selection for task-specific model finetuning. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-0325. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/13848b5893119ff772b69812c95914fa-Abstract-Conference.html.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. The Flan collection: Designing data and methods for effective instruction tuning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 22631–22648. PMLR, 2023. URL https://proceedings.mlr.press/v202/longpre23a.html.

Da Ma, Gonghu Shang, Zhi Chen, Libo Qin, Yijie Luo, Hongshen Xu, Lei Pan, Shuai Fan, Kai Yu, and Lu Chen. Task-specific data selection for instruction tuning via monosemantic neuronal activations. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 138006–138034. Curran Associates, Inc., 2025. doi: 10.52202/085713-4607. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/c971ffd786876d5927caf482b06bcba5-Paper-Conference.pdf.

Guanghui Min, Tianhao Huang, Ke Wan, and Chen Chen. GIST: Targeted data selection for instruction tuning via coupled optimization geometry. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=K2KeBuKAtu.

David Mueller, Mark Dredze, and Nicholas Andrews. Multi-task transfer matters during instruction-tuning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 14880–14891. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-acl.883. URL https://aclanthology.org/2024.findings-acl.883/.

Nihal V. Nayak, Paula Rodriguez-Diaz, Neha Hulkund, Sara Beery, and David Alvarez-Melis. A critical look at targeted instruction selection: Disentangling what matters (and what doesn't). In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=Dy5GeCd003.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2362–2376. Association for Computational Linguistics, 2020.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Noam Shazeer. GLU variants improve transformer, 2020. URL https://arxiv.org/abs/2002.05202.

Ran Song, Shizhu He, Shuting Jiang, Yantuan Xian, Shengxiang Gao, Kang Liu, and Zhengtao Yu. Does large language model contain task-specific neurons? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 7101–7113. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.emnlp-main.403. URL https://aclanthology.org/2024.emnlp-main.403/.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 13003–13051. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-acl.824. URL https://aclanthology.org/2023.findings-acl.824/.

Shaomu Tan, Di Wu, and Christof Monz. Neuron specialization: Leveraging intrinsic task modularity for multilingual machine translation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 6506–6527, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.374. URL https://aclanthology.org/2024.emnlp-main.374/.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. Language-specific neurons: The key to multilingual capabilities in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5701–5715. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.309. URL https://aclanthology.org/2024.acl-long.309/.

Jingtan Wang, Xiaoqiang Lin, Rui Qiao, Pang Wei Koh, Chuan-Sheng Foo, and Bryan Kian Hsiang Low. NICE data selection for instruction tuning in LLMs with non-differentiable evaluation metric. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 63662–63689. PMLR, 2025a. URL https://proceedings.mlr.press/v267/wang25bm.html.

Weixuan Wang, Barry Haddow, Minghao Wu, Wei Peng, and Alexandra Birch. Sharing matters: Analysing neurons across languages and tasks in LLMs, 2025b. URL https://arxiv.org/abs/2406.09265.

Xiaozhi Wang, Kaiyue Wen, Zhengyan Zhang, Lei Hou, Zhiyuan Liu, and Juanzi Li. Finding skill neurons in pre-trained transformer-based language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 11132–11152. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.emnlp-main.765. URL https://aclanthology.org/2022.emnlp-main.765/.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 95266–95290. Curran Associates, Inc., 2024. doi: 10.52202/079017-3018. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/ad236edc564f3e3156e1b2feafb99a24-Paper-Datasets_and_Benchmarks_Track.pdf.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022a.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, 2022b. doi: 10.52202/068431-1800.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. BloombergGPT: A large language model for finance, 2023. URL https://arxiv.org/abs/2303.17564.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. LESS: Selecting influential data for targeted instruction tuning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 54104–54132. PMLR, 2024. URL https://proceedings.mlr.press/v235/xia24c.html.

Tingyu Xia, Bowen Yu, Kai Dang, An Yang, Yuan Wu, Yuan Tian, Yi Chang, and Junyang Lin. Rethinking data selection at scale: Random selection is almost all you need. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 2698–2711. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-emnlp.146. URL https://aclanthology.org/2025.findings-emnlp.146/.

Yu Yang, Siddhartha Mishra, Jeffrey Chiang, and Baharan Mirzasoleiman. SmallToLarge (S2L): Scalable data selection for fine-tuning large language models by summarizing training trajectories of small models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 83465–83496. Curran Associates, Inc., 2024. doi: 10.52202/079017-2655. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/97fe251c25b6f99a2a23b330a75b11d4-Paper-Conference.pdf.

Yuan Zhang, Lifeng Guo, Junwen Pan, Wenzhao Zheng, Wen Zhou, Kuan Cheng, Kurt Keutzer, and Shanghang Zhang. SEED: Targeted data selection by weighted independent set, 2026. URL https://arxiv.org/abs/2605.15691.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. LlamaFactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pp. 400–410, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-demos.38. URL https://aclanthology.org/2024.acl-demos.38/.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. LIMA: Less is more for alignment. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

A Experimental Details

This section provides additional details on the reference sets, training configuration, evaluation protocol, and baseline implementation used throughout our experiments. These settings are kept consistent across methods whenever applicable to ensure that performance differences mainly reflect the selected training data.

A.1 Reference and Background Sets

We use a small reference set for each reference task to construct task-level signals for data selection. All methods that require reference data use the same reference sets, which controls for differences in reference-set composition and enables a consistent comparison across selection methods. For MMLU, GSM8K, and TyDiQA, we use 100 randomly sampled reference examples. For BBH, we use all 81 examples from the official 3-shot demonstrations, while for MBPP, we use all 90 examples from the validation set. In addition, STAR uses a background set to normalize neuron activation statistics before constructing the task profiles. We randomly sample 500 examples from the candidate pool to form this background set, which provides a common reference distribution for activation normalization across all reference tasks. Table 6 summarizes the sources and construction of the reference sets.

A.2 STAR Implementation Details

For the Task-Shared score, we set $\beta = 0.5$ in $S^{\mathrm{sh}}(x) = \mathrm{Mean}_m\, q_m(x) - \beta\, \mathrm{Std}_m\, q_m(x)$, such that variation in candidate alignment across the reference-task prototypes is penalized by half of its standard deviation.

For cluster-aware subset construction, we apply MiniBatchKMeans with $C = 100$ clusters, a batch size of 4,096, and random seed 42. Clustering is performed on the L2-normalized concatenation of the weighted Task-Shared and Target-Specific activation representations. Given a selection budget B, we assign each cluster a base quota of $q = \lfloor B/C \rfloor$ and retain up to q examples with the highest final STAR scores within each cluster. If the resulting subset contains fewer than B examples, either because of the remainder in the budget or because some clusters contain fewer than q examples, the remaining slots are filled globally with the highest-scoring unselected candidates.

For activation extraction, we concatenate the prompt and response and mean-pool activation features over all non-padding tokens in the resulting sequence. We use a maximum sequence length of 2,048 for activation-based data selection. This token aggregation differs from the supervised fine-tuning objective: the response-only loss mask is applied during subsequent SFT, but is not used when extracting activation features for data selection.

A.3 Training and Evaluation Details

We conduct the main experiments using Llama-3.1-8B and additionally evaluate STAR on Qwen2.5-7B. All experiments, including data selection, model fine-tuning, and evaluation, are conducted on NVIDIA H100 GPUs. For both backbones, we perform full-parameter supervised fine-tuning with LLaMA-Factory for two epochs using four H100 GPUs and DeepSpeed ZeRO-2. We use a maximum sequence length of 4,096, bfloat16 precision, a per-device batch size of 1, and 32 gradient-accumulation steps, resulting in a global batch size of 128. The learning rate is 7×10^{-6} with a cosine scheduler, a warmup ratio of 0.01, weight decay of 0.1, and gradient clipping at 1.0. The training loss is computed only over assistant-response tokens. All data-selection methods use the same training configuration for a given backbone.

We use the corresponding instruction template for each backbone: `llama3` for Llama-3.1-8B and `qwen` for Qwen2.5-7B. Figure 4 illustrates the input formatting with the same instruction-response example for both backbones. Llama-3.1-8B represents the user and assistant messages using its header and end-of-turn tokens, while Qwen2.5-7B follows its chat template with system, user, and assistant messages. These backbone-specific templates are applied consistently during fine-tuning.

The 5% and 1% selection budgets correspond to 13,534 and 2,706 training examples, respectively. Llama-3.1-8B serves as the primary backbone and is evaluated on all five reference-task benchmarks

Table 6: Reference sets used for data selection. MMLU, GSM8K, and TyDiQA use 100 randomly sampled reference examples, while BBH and MBPP use all 81 official 3-shot demonstrations and all 90 validation examples, respectively.

Reference Task	Reference Set Source	Construction	# Examples
MMLU	MMLU validation	Random sampling	100
BBH	Official 3-shot demonstrations	All examples	81
GSM8K	GSM8K train	Random sampling	100
TyDiQA	TyDiQA-GoldP train	Random sampling	100
MBPP	MBPP validation	All examples	90

Table 7: Evaluation settings for the reference-task and OOD benchmarks.

Benchmark	Evaluation Setting	Metric	Implementation
MMLU	5-shot	Accuracy	`lm-evaluation-harness`
BBH	3-shot CoT	Exact Match	`lm-evaluation-harness`
TyDiQA	1-shot	F1	LESS codebase
GSM8K	8-shot CoT	Exact Match	`lm-evaluation-harness`
MBPP	3-shot	Pass@1	`lm-evaluation-harness`
MMLU-Pro	5-shot	Accuracy	`lm-evaluation-harness`
XCOPA	0-shot	Accuracy	`lm-evaluation-harness`
MATH	4-shot	Exact Match	`lm-evaluation-harness`
HumanEval	0-shot	Pass@1	`lm-evaluation-harness`

and four OOD benchmarks. Qwen2.5-7B is used for the additional-backbone experiment and follows the same training configuration and target-task evaluation settings.

We use `lm-evaluation-harness` for all benchmarks except TyDiQA, for which we use the evaluation implementation from the LESS codebase. We follow a fixed evaluation setting for each benchmark across all methods to ensure direct comparability. All scores are reported as percentages. Table 7 summarizes the evaluation settings and metrics.

A.4 Baseline Implementation

We compare STAR with LESS, MONA, NAIT, and BIDS. We reproduce all baseline methods using their released code when accessible; otherwise, we implement them following the settings described in the corresponding papers. All methods use the same candidate pool and selection budgets, and methods that require reference data use the same reference sets. After data selection, all selected subsets are fine-tuned using the same training configuration for a given backbone. These controls reduce differences caused by data, reference-set, or optimization settings and allow the comparison to focus on the data-selection strategies.

B Additional Results

B.1 Detailed Results

Table 8 reports the complete task-level results under the 5% selection budget. Ref. Avg. denotes the average performance across the five reference-task benchmarks for each target setting, while OOD Avg. denotes the average across the four OOD benchmarks for the same trained model. STAR achieves the highest Ref. Avg. in all five target settings, showing that its target-specific selection generally retains strong performance across the full set of reference tasks. For OOD evaluation, STAR also achieves the highest OOD Avg. in all five target settings. Overall, these task-level results provide a more detailed view of the variation across target settings and complement the aggregate Target Avg., Non-Target Avg., Aligned OOD Avg., and Cross-Target OOD Avg. reported in the main text.

Llama-3.1-8B

<|begin_of_text|><|start_header_id|>user<|end_header_id|>

Denny asked: Premise: "A man in a full wetsuit surfs a crashing wave."
Based on this premise, can we conclude that the hypothesis "A man surfing near hawaii." is true?
Options:
- yes
- it is not possible to tell
- no.
Le: OK, so how can I answer with some explanation?
<|eot_id|><|start_header_id|>assistant<|end_header_id|>

Hawaii is not the only place where crashing waves can be surfed.
The answer: it is not possible to tell.<|eot_id|>

Qwen2.5-7B

<|im_start|>system
You are Qwen, created by Alibaba Cloud. You are a helpful assistant.<|im_end|>
<|im_start|>user
Denny asked: Premise: "A man in a full wetsuit surfs a crashing wave."
Based on this premise, can we conclude that the hypothesis "A man surfing near hawaii." is true?
Options:
- yes
- it is not possible to tell
- no.
Le: OK, so how can I answer with some explanation?
<|im_end|>
<|im_start|>assistant
Hawaii is not the only place where crashing waves can be surfed.
The answer: it is not possible to tell.<|im_end|>

Figure 4: Example input formatting used for instruction tuning with Llama-3.1-8B and Qwen2.5-7B. The same instruction-response example is formatted using the corresponding instruction template of each backbone.

B.2 Comparison with Multi-Reference Baselines

We extend MONA and NAIT to use the same five reference tasks as STAR. For each baseline, we compute a task-specific candidate score $S_m(x)$ for each reference task m and aggregate the scores from the non-target tasks as

$$S_{-t}(x) = \frac{1}{M-1} \sum_{m \neq t} S_m(x). \tag{11}$$

We then combine the designated-target score and the aggregated non-target score:

$$S_{\text{multi}}(x, t) = (1 - \lambda) S_t(x) + \lambda S_{-t}(x), \tag{12}$$

where λ controls the relative contribution of the target and non-target reference tasks. We set $\lambda = 0.5$ in our experiments.

Table 8: Detailed results under the 5% selection budget. Ref. Avg. averages the five reference-task benchmarks within each row, and OOD Avg. averages the four OOD benchmarks. Within each target setting, the best and second-best results in each column are bolded and underlined, respectively.

		Reference-Task Benchmarks						OOD Benchmarks				
Target	**Method**	**MMLU**	**BBH**	**TyDiQA**	**GSM8K**	**MBPP**	**Ref. Avg.**	**MMLU-Pro**	**XCOPA**	**MATH**	**H-Eval**	**OOD Avg.**
MMLU	LESS	62.63	60.65	68.72	42.08	46.60	56.14	34.45	62.42	14.20	35.37	36.61
	MONA	63.58	64.17	69.40	41.09	46.60	56.97	34.64	63.82	16.38	39.63	38.62
	NAIT	64.24	63.33	**72.15**	48.14	45.80	58.73	**37.30**	62.56	15.72	40.24	38.96
	STAR	**64.81**	**64.44**	69.81	**52.16**	**47.00**	**59.64**	37.28	**63.85**	**16.58**	**41.46**	**39.79**
BBH	LESS	62.68	63.89	71.72	**56.18**	48.20	60.53	34.23	62.69	16.42	42.07	38.85
	MONA	63.20	64.26	63.57	54.21	**49.20**	58.89	34.46	63.22	16.42	40.24	38.59
	NAIT	**64.91**	64.17	69.39	55.34	46.40	60.04	35.61	61.96	**17.26**	40.85	38.92
	STAR	64.39	**65.65**	**72.16**	55.42	48.60	**61.24**	**35.66**	**63.29**	16.90	**42.68**	**39.63**
TyDiQA	LESS	63.30	59.81	71.71	41.70	47.20	56.74	33.75	62.78	14.50	**42.68**	38.43
	MONA	**64.73**	**62.78**	**72.26**	53.68	**48.20**	60.33	**36.79**	62.93	5.90	40.24	36.47
	NAIT	63.52	61.48	71.45	56.03	47.20	59.94	36.18	62.35	17.20	40.85	39.14
	STAR	64.60	62.31	72.01	**56.33**	47.80	**60.61**	36.59	**63.25**	**18.28**	**42.68**	**40.20**
GSM8K	LESS	63.45	63.70	**68.74**	58.00	46.80	60.14	36.48	62.58	19.24	43.29	40.40
	MONA	63.30	62.04	63.65	60.42	46.40	59.16	36.59	**65.75**	17.42	42.07	40.46
	NAIT	63.67	63.43	68.14	55.50	**48.40**	59.83	36.43	62.87	18.76	41.46	39.88
	STAR	**64.25**	**64.72**	67.43	**62.93**	**48.40**	**61.55**	**36.93**	62.64	**19.94**	**43.90**	**40.85**
MBPP	LESS	60.55	60.93	71.12	52.31	47.00	58.38	35.46	62.58	13.94	39.63	37.90
	MONA	64.31	63.98	70.19	52.69	47.40	59.71	**36.34**	58.76	16.78	**44.51**	39.10
	NAIT	63.55	**65.46**	70.04	46.63	48.20	58.78	33.51	**63.16**	16.58	40.24	38.37
	STAR	**64.76**	63.15	**71.13**	**55.34**	**48.60**	**60.60**	35.95	62.89	**17.04**	43.90	**39.95**

As shown in Table 9, STAR achieves the best target-task performance in all five target settings. For cross-task utility, STAR attains the highest Non-Target Avg. in three of the five settings (MMLU, BBH, and MBPP) and ranks second in the remaining two, where NAIT-5ref is slightly higher (TyDiQA and GSM8K). For OOD transfer, STAR achieves the highest OOD Avg. in four of the five settings and is within 0.1 points of the best result on GSM8K. Using multiple reference tasks improves broader task utility for MONA and NAIT, while STAR retains stronger target and OOD performance in most settings. This comparison highlights the benefit of explicitly separating and combining Task-Shared and Target-Specific signals.

C Additional Ablation Studies

C.1 Sensitivity to the Number of Selected Neurons

We examine the sensitivity of STAR to the number of selected neurons by varying both K^{sh} and K^{sp} over $\{3072, 4096, 5120\}$ while keeping all other settings unchanged. This analysis evaluates the robustness of STAR to changes in the number of selected neurons around the default configuration used in the main experiments. Table 10 reports the target-task performance under the 5% selection budget.

Performance varies only modestly across the tested neuron counts. The default configuration with 4096 neurons achieves a Target Avg. of 62.80, compared with 62.37 and 61.77 for 3072 and 5120 neurons, respectively. These results indicate that the performance of STAR remains stable under moderate changes in the number of selected neurons.

C.2 Contributions of Neuron Selection and Clustering

We examine the contribution of clustering by removing only the clustering step while keeping the neuron selection, candidate scoring, selection budget, and training configuration unchanged. Together with the neuron-selection ablation in Table 5, this comparison distinguishes the roles of activation subspace design and cluster-aware subset construction. The neuron-selection results show that the dedicated Task-Shared and Target-Specific subspaces improve all three aggregate metrics over all-

Table 9: Comparison with multi-reference baselines under the 5% selection budget across all five target settings. All methods use the same five reference tasks. *Target* reports performance on the designated target task, and *Non-Target Avg.* averages the remaining four reference tasks. *Aligned OOD* reports the OOD benchmark from the predefined capability pairings (MMLU–MMLU-Pro, TyDiQA–XCOPA, GSM8K–MATH, and MBPP–HumanEval); BBH has no aligned OOD benchmark. *OOD Avg.* averages the four OOD benchmarks for the model trained in each target setting. Within each target setting, the best and second-best results in each column are bolded and underlined, respectively.

Target Task	Method	Target	Non-Target Avg.	Aligned OOD	OOD Avg.
MMLU	MONA-5ref	64.35	57.68	<u>37.12</u>	<u>38.91</u>
	NAIT-5ref	<u>64.40</u>	<u>58.08</u>	36.73	38.87
	STAR	**64.81**	**58.35**	**37.28**	**39.79**
BBH	MONA-5ref	<u>64.81</u>	<u>59.02</u>	—	37.96
	NAIT-5ref	63.52	58.76	—	<u>38.72</u>
	STAR	**65.65**	**60.14**	—	**39.63**
TyDiQA	MONA-5ref	70.97	57.21	<u>62.07</u>	<u>40.07</u>
	NAIT-5ref	<u>71.26</u>	**58.02**	61.38	39.29
	STAR	**72.01**	<u>57.76</u>	**63.25**	**40.20**
GSM8K	MONA-5ref	<u>61.18</u>	60.29	17.20	39.27
	NAIT-5ref	60.05	**62.37**	<u>18.72</u>	**40.92**
	STAR	**62.93**	<u>61.20</u>	**19.94**	<u>40.85</u>
MBPP	MONA-5ref	<u>48.20</u>	<u>63.39</u>	43.29	38.66
	NAIT-5ref	<u>48.20</u>	62.78	**44.51**	<u>39.28</u>
	STAR	**48.60**	**63.60**	<u>43.90</u>	**39.95**

Table 10: Effect of the number of selected neurons under the 5% selection budget. The same number of neurons is used for the Task-Shared and Target-Specific sets.

$K^{\text{sh}} = K^{\text{sp}}$	MMLU	BBH	TyDiQA	GSM8K	MBPP	Target Avg.
3072	64.09	64.91	71.95	**63.31**	47.60	62.37
4096	**64.81**	**65.65**	**72.01**	62.93	**48.60**	**62.80**
5120	64.34	65.28	68.71	62.32	48.20	61.77

neuron or randomly selected FFN spaces. Table 11 further shows that clustering yields additional gains with the activation-based scoring fixed. The larger improvement in Non-Target Avg. suggests that clustering complements the relevance signals by improving the coverage of the selected subset. Overall, the two ablations show that activation subspace design and cluster-aware subset construction make complementary contributions to the final selection procedure.

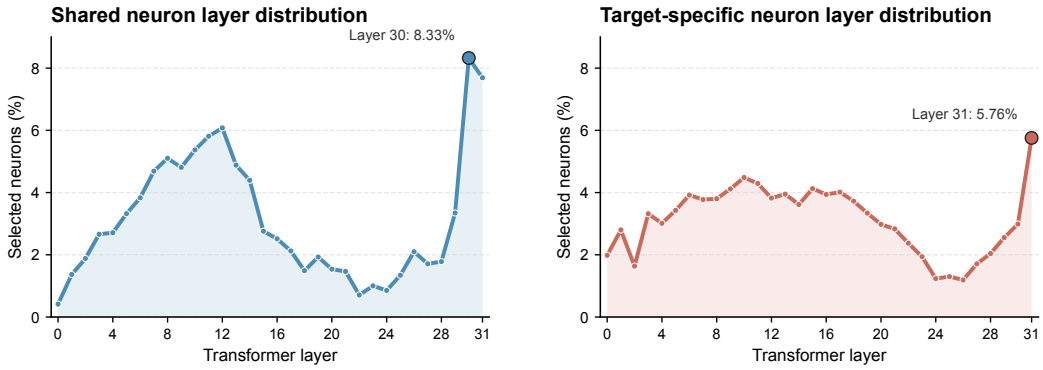
D Additional Analysis

D.1 Layer Distribution of Selected Neurons

We examine the layer distribution of the neurons selected by STAR to better understand where the two activation-defined neuron sets occur within the model. As shown in Figure 5, both Task-Shared and Target-Specific neurons are distributed across multiple Transformer layers rather than being concentrated in a single layer. The two neuron sets nevertheless exhibit different layer-wise distributions. Task-Shared neurons occur at relatively high proportions in several middle layers and increase sharply in the final layers, with the highest proportion of 8.33% at Layer 30. Target-Specific neurons are distributed across a broader range of layers and reach their highest proportion of 5.76% at Layer 31. Thus, both signals involve neurons from different depths of the network, while the relative concentration across layers differs between the two sets. Overall, the two selection criteria yield distinct layer-wise distributions across the model.

Table 11: Effect of clustering under the 5% selection budget. All other components of STAR are kept unchanged.

Variant	Target Avg. ↑	Non-Target Avg. ↑	Cross-Target OOD Avg. ↑
w/o Clustering	62.67	59.39	39.87
STAR	**62.80**	**60.21**	**40.08**



(a) Task-Shared neurons.

(b) Target-Specific neurons.

Figure 5: Layer distribution of the neurons selected by STAR. The y-axis reports the percentage of selected neurons in each Transformer layer.

D.2 Activation Patterns of Selected Neurons

We further visualize the cross-task activation patterns of representative Task-Shared, Target-Specific, and unselected neurons in Figure 6. The three groups exhibit clearly different activation patterns across the reference tasks. Task-Shared neurons generally exhibit positive activation across multiple reference tasks, consistent with their selection based on high and relatively balanced cross-task activation. This pattern indicates that their selection is not dominated by a single reference task. In contrast, Target-Specific neurons exhibit stronger activation for their corresponding target tasks than for the remaining reference tasks, producing a clear target-dependent pattern that is consistent with their selection based on relative target activation. The displayed unselected neurons show more heterogeneous activation patterns without the same cross-task consistency or target-specific structure. The contrast among the three groups provides a direct visualization of the activation statistics used by STAR and supports the use of Task-Shared and Target-Specific neurons as complementary views for data selection.

D.3 Source Composition of Selected Subsets

We examine the composition of the selected subsets with respect to the four sources of the candidate pool: FLAN v2, Chain-of-Thought (CoT), Dolly, and OpenAssistant. As shown in Figure 7, the subsets selected by STAR remain close to the pool composition under every designated target, with only moderate target-dependent reweighting.

D.4 Qualitative Analysis of Selected Examples

We qualitatively analyze the selected subsets with the assistance of an LLM (Claude Fable 5), which inspects the top-ranked examples of each method for the MMLU target under the 5% selection budget; representative cases are shown in Table 12. The top-ranked examples of LESS are long reading-comprehension passages answered with a single option letter: they match the multiple-choice format of MMLU but carry little transferable content, and their influence attribution points to unrelated subtasks, suggesting that gradient-based influence is dominated by the output format. MONA's top-ranked examples are knowledge-dense extractive QA pairs, including a CoT example with an

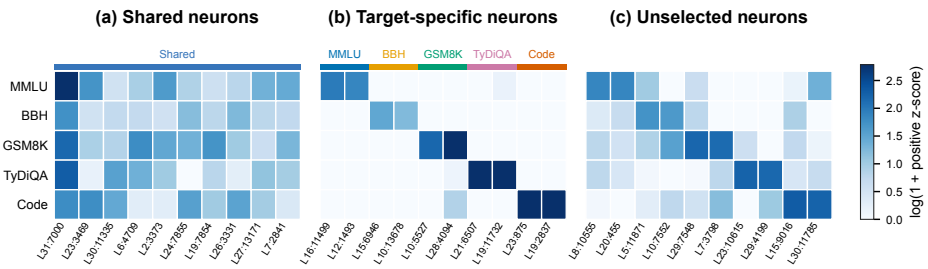


Figure 6: Cross-task activation patterns of representative Task-Shared, Target-Specific, and unselected neurons. Each cell shows $\log(1 + \text{positive z-score})$ for a neuron on a reference task. Task-Shared neurons exhibit positive activation across multiple reference tasks, while Target-Specific neurons exhibit stronger activation for their corresponding target tasks.

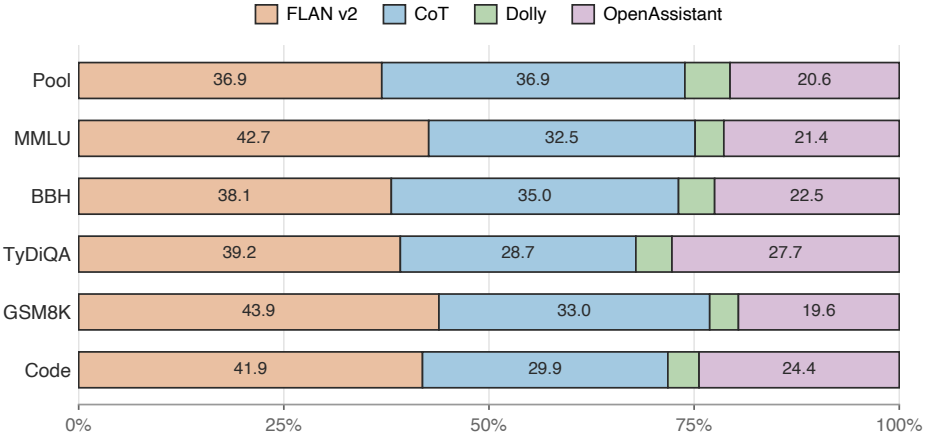


Figure 7: Source composition of the subsets selected by STAR under the 5% selection budget across the five target settings. The top bar shows the composition of the full candidate pool as a reference.

inverted supervision format whose response generates a new question rather than answering one, indicating that a single prototype-similarity signal ignores the instruction–response direction.

In contrast, the STAR examples score high under both signals: a multiple-choice combinatorics example with an explicit reasoning step, and a multi-turn Spanish conversation that generates and grades quiz questions. Both are MMLU-like in form yet broadly useful in content, which is precisely what a single target-similarity score cannot identify.

Table 12: Representative examples selected for the MMLU target under the 5% selection budget. For LESS and MONA we show top-ranked examples under their own selection scores; for STAR we show examples whose Task-Shared and Target-Specific percentile ranks are both high. Texts are truncated.

Example	Content (truncated)
LESS (top-ranked by influence score)	
`flan_v2_40417` rank 1, influence 0.197	*User:* Read the article and select the best answer. Article: For many years scientists have said that the earth is getting warmer. Now they have found something new—hot cities! [...] *Assistant:* A
`flan_v2_56882` rank 3, influence 0.172	*User:* Read the article and select the best answer. Article: A serious problem for today's society is who should be responsible for our elderly [...] *Assistant:* C
MONA (top-ranked by prototype-similarity score)	
`flan_v2_10241` rank 1, score 0.305	*User:* Answer a question about this article: Depending on a number of considerations, a company may apply for and be granted a patent [...] *Assistant:* cotton fiber paper
`cot_74890` rank 14, score 0.297	*User:* Ans: Dmitri Mendeleev. Ques: The Russian chemist Dmitri Mendeleev was the first scientist to make a periodic table [...] *Assistant:* Ques: Passage: Webbed toes is the common name for syndactyly [...] Question: Based on this passage, what is the medical term for webbed toes? CoT: The relevant information is [...]
STAR (Task-Shared and Target-Specific percentile ranks both high)	
`cot_17683` $\widehat{S}^{\mathrm{sh}}$=0.99, $\widehat{S}^{\mathrm{sp}}$=0.99	*+User:* Question. Five coins are tossed simultaneously. In how many of the outcomes will the third coin turn up a head? Options: (A) 2^4 [...] Answer. (A) What step-by-step reasoning justifies that answer? *Assistant:* Reasoning: Fix the third coin as H. The remaining 4 coins have 2^4 outcomes.
`oasst1_36653` $\widehat{S}^{\mathrm{sh}}$=0.97, $\widehat{S}^{\mathrm{sp}}$=0.97	*User (Spanish):* Hazme preguntas tipo test sobre el universo con 3 respuestas (a, b, c) donde solo una de ellas es correcta [...] *Assistant:* 1. El sistema solar esta formado por los 8 planetas [...] a) Mercurio, Venus, Tierra, Marte [...] 2. ¿Qué gigante gaseoso posee un periodo de rotación de 10 horas? [...]