SVDoRA: Spectral Verification-Aware Direction-Oriented Rank Adaptation via Outcome-Guided Spectral Annealing

Anonymous authors
Paper under double-blind review

Abstract

SVD-based parameter-efficient fine-tuning selects spectral directions by correctness-agnostic scoring criteria and treats every selected direction as a generation update. We propose **SVDoRA**, a *verification-aware* adapter that uses gold-outcome labels to allocate capacity across a residual subspace and two spectral roles: task-specific generation directions (TSGD) and task-specific verification directions (TSVD). After a short residual-adapter warmup, a one-time calibration pass regenerates answers, labels them against gold outcomes, and selects the two roles by outcome-guided spectral annealing; the chosen directions then stay fixed for the rest of training. Compared with standard rank-r LoRA, SVDoRA trains r residual singular values and $k_g + k_v$ direction coefficients in addition to the two residual factors—24 extra scalars per target matrix at the canonical $r = 16$, $k_g = k_v = 4$ setting. SVDoRA reaches 88.1% average accuracy on Qwen2.5-7B commonsense reasoning at rank 16 (+1.4 pp over LoRA-DASH) and 79.0% at 1.5B rank 16; it leads the Llama-3 GSM8K+MATH average (48.4%), with competitive transfer to factual QA and code generation.

1 Introduction

Fully fine-tuning large language models (LLMs) such as Qwen2.5 (Qwen Team, 2025a), Qwen3 (Qwen Team, 2025b), and DeepSeek-R1 (DeepSeek-AI, 2025) is prohibitively expensive at billion-parameter scale, and LoRA (Hu et al., 2022), which learns a low-rank additive update $\Delta\mathbf{W} = \mathbf{BA}$ to frozen pretrained weights, has become the dominant parameter-efficient fine-tuning (PEFT) paradigm. A growing line of work initializes such adapters from the singular value decomposition (SVD) of the pretrained weights: PiSSA (Meng et al., 2024) adapts principal components, MiLoRA (Wang et al., 2025) minor ones, SORSA (Cao & Song, 2025) adds orthonormal regularization, LoRA-Dash (Si et al., 2025) formally defines *Task-Specific Directions* (TSDs), and DisLoRA (Yifei et al., 2025) selects the directions most aligned with residual weight changes; their consistent gains over LoRA show that the weight spectrum is a meaningful prior for adapter design.

Despite these advances, existing SVD-based PEFT methods share two fundamental limitations. *First*, prior SVD-based PEFT has not tested whether outcome supervision can assign distinct adaptation roles to selected spectral subsets. Kadavath et al. (Kadavath et al., 2022) elicit prompted confidence in answer correctness, while Weng et al. (Weng et al., 2023) use inference-time backward verification to score candidate answers; both are output-space settings distinct from our one-time training-stage selection. *Second, direction selection is driven by scoring criteria that are never grounded in whether the model actually answers correctly*: spectral alignment (Si et al., 2025; Yifei et al., 2025), gradient magnitude, or entropy-based uncertainty, none of which observes a single ground-truth correctness label. Motivated by outcome-verified rewards in RISE (Liu et al., 2025), we use gold-answer correctness to supervise weight-space direction selection. That check has to be automatic: an exact match, a multiple-choice letter, or a unit test. For code generation, we take a shortcut at calibration: using text similarity instead of code execution, because running unit tests in that one-shot pass would slow the Phase 2 of fine-tuning down.

We address both limitations through **SVDoRA**: *Spectral Verification-Aware Direction-Oriented Rank Adaptation via Outcome-Guided Spectral Annealing*. It partitions initial spectral indices into

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.

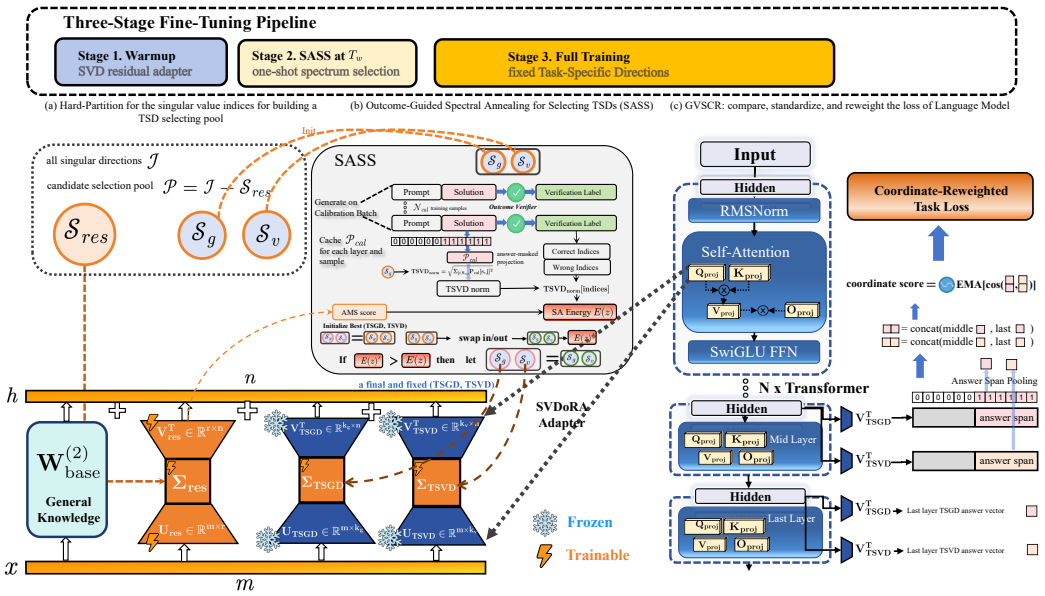

Figure 1: **Overview of SVDoRA. Top:** The three-stage pipeline selects direction sets once at step T_w and freezes them thereafter. **(a)** The hard partition: $\mathcal{S}_{\text{res}}$ lies outside the candidate pool $\mathcal{P}$, while TSGD ($\mathcal{S}_{\text{gen}}$) and TSVD ($\mathcal{S}_{\text{ver}}$) are disjoint subsets inside $\mathcal{P}$; only the residual factors and the scalar coefficients are trainable. **(b)** SASS selects $(\mathcal{S}_{\text{gen}}, \mathcal{S}_{\text{ver}})$ by outcome-guided spectral annealing over verifier-labeled calibration samples (Figure 2 and Algorithm 1). **(c)** GVSCR reweights samples using an EMA-standardized, coordinate-dependent TSGD–TSVD cosine; the detached coordinate branch is not a calibrated confidence measure (Algorithm 2).

a frozen remainder, a residual adapter, and two outcome-supervised roles on disjoint spectral index sets, termed TSGD and TSVD. We call this *verification-aware capacity allocation*: correctness labels influence which fixed spectral directions receive trainable coefficients. After warmup, a one-time annealing pass (Kirkpatrick et al., 1983) selects the two sets and freezes them. The method performs no inference-time error detection, revision, or regeneration.

Empirically, SVDoRA attains the highest commonsense averages at rank r=16 on both Qwen2.5-1.5B and 7B, and remains competitive on mathematical reasoning, code generation, and factual QA. Linear probes on Qwen2.5-1.5B math further show that the selected TSVD features linearly decode answer correctness at the dimension the method actually deploys.

The contributions of this work are:

- A **four-way partition of the initial spectral indices** with outcome-supervised TSGD/TSVD adaptation roles.
- An **outcome-guided spectral annealing** procedure that uses verifier-labelled calibration without an extra trainable verifier.
- Evidence across four task families and three base models, with a deliberately bounded, non-exhaustive probing analysis.

Beyond the SVD-based adapters surveyed above, related work spans self-verification and uncertainty estimation in LLMs, output-space RL methods that exploit verification signals such as RISE (Liu et al., 2025) and InSeV (Zhang et al., 2025), and correctness-agnostic selection criteria paired with combinatorial search. We defer the full survey to §A in the supplementary material.

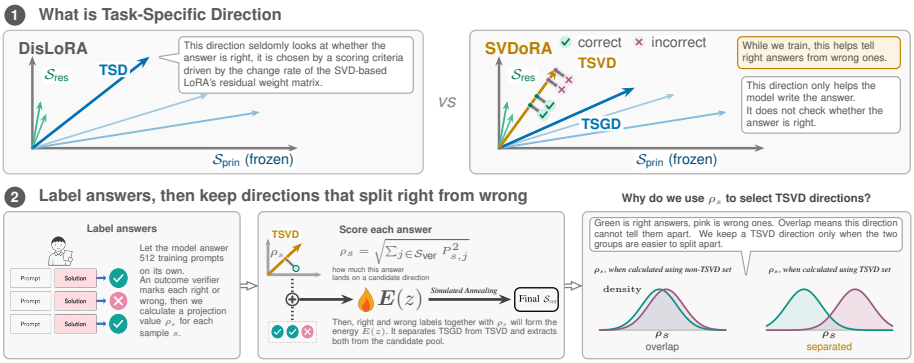

Figure 2: **Outcome-verified calibration (SASS). Top:** DisLoRA ranks its TSD by the change rate of the LoRA residual weights; SVDoRA selects TSGD ($\mathcal{S}_{\text{gen}}$) and TSVD ($\mathcal{S}_{\text{ver}}$) by annealing, choosing TSVD to separate right from wrong answers. **Bottom:** A verifier will label the model's fresh answers to 512 training prompts as right or wrong. SASS then projects answers onto candidate directions and those that can split the right and wrong sample groups are kept as TSVDs; the Metropolis search of Figure A1 minimizes the selection energy.

2 Method

2.1 Spectral Decomposition Framework

For each target weight matrix $\mathbf{W}_0 \in \mathbb{R}^{m\times n}$, we compute $\mathbf{W}_0 = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top = \sum_{i=1}^{\min(m,n)} \sigma_i \mathbf{u}_i \mathbf{v}_i^\top$, where $\sigma_1 \geq \sigma_2 \geq \cdots \geq 0$. Let $\mathcal{I} = \{1, \ldots, \min(m,n)\}$ index all initial singular directions. Let $\mathcal{S}_{\text{res}}$ be the bottom-r initialization indices and $\mathcal{P} = \mathcal{I} \setminus \mathcal{S}_{\text{res}}$ the candidate pool. SASS selects disjoint $\mathcal{S}_{\text{gen}}, \mathcal{S}_{\text{ver}} \subset \mathcal{P}$, and we define $\mathcal{S}_{\text{freeze}} = \mathcal{I} \setminus (\mathcal{S}_{\text{res}} \cup \mathcal{S}_{\text{gen}} \cup \mathcal{S}_{\text{ver}})$. Thus $\mathcal{S}_{\text{freeze}}, \mathcal{S}_{\text{res}}, \mathcal{S}_{\text{gen}}, \mathcal{S}_{\text{ver}}$ partition the initial spectral indices:

- $\mathcal{S}_{\text{freeze}}$: unselected directions whose contribution remains in the frozen implementation base tensor.
- $\mathcal{S}_{\text{res}}$: bottom-r directions used to initialize the trainable residual adapter.
- $\mathcal{S}_{\text{gen}}$ (TSGD): k_g task-specific generation directions selected from $\mathcal{P}$.
- $\mathcal{S}_{\text{ver}}$ (TSVD): k_v task-specific verification directions selected from $\mathcal{P} \setminus \mathcal{S}_{\text{gen}}$.

Figure 1 summarizes this partition and the post-selection training paths. Drawing task directions only from $\mathcal{P}$ is a **hard partition** that keeps them disjoint from the residual subspace, which makes the orthogonality penalty of §2.4 well-posed rather than a constraint on partially overlapping subspaces. These four sets only assign starting directions. The residual adapter is then a free product $R(t) = U_{\text{res}}(t)\,\text{diag}(s_{\text{res}}(t))V_{\text{res}}^\top(t)$: all three factors stay trainable and are not pinned to the original bottom-r subspace. As in LoRA, we scale every adapter term by $s = \alpha_{\text{LoRA}}/r$ (canonically $s = 2$). Selected directions receive trainable scalar coefficients ($\{\alpha_i(t)\}_{i=1}^{k_g}$, $\{\beta_j(t)\}_{j=1}^{k_v}$), while their rank-1 structure stays frozen. After extraction, the implemented weight is

$$\mathbf{W}(t) = \underbrace{\mathbf{W}_{\text{base}}^{(2)}}_{\text{frozen implementation base}} + s \underbrace{R(t)}_{\text{residual adapter}} + s \underbrace{\sum_{i\in\mathcal{S}_{\text{gen}}} \alpha_i(t)\,\mathbf{u}_i\mathbf{v}_i^\top}_{\text{TSGD}} + s \underbrace{\sum_{j\in\mathcal{S}_{\text{ver}}} \beta_j(t)\,\mathbf{u}_j\mathbf{v}_j^\top}_{\text{TSVD}}. \tag{1}$$

The two task-direction sums share this rank-1 form; they are distinguished by the outcome-guided selection energy in §2.2. Because we scale every adapter term by s and later subtract the extracted directions, $\mathbf{W}_{\text{base}}^{(2)}$ is only the frozen remainder we keep, rather than the unscaled sum over $\mathcal{S}_{\text{freeze}}$.

2.2 OUTCOME-GUIDED DIRECTION SELECTION

Where prior SVD-based methods rank directions by correctness-agnostic scoring criteria, SVDoRA selects $(\mathcal{S}_{\text{gen}}, \mathcal{S}_{\text{ver}})$ *jointly* by simulated annealing over an *outcome-verified* energy.

Generation alignment (AMS). Each candidate direction is scored by an **Alignment Magnitude Score** (AMS_i), inherited from the "change rate" criterion of DisLoRA (Yifei et al., 2025) and LoRA-Dash (Si et al., 2025) (formula in Eq. (A1), §B.4). High-AMS directions are highly task-adaptive and therefore tightly coupled to the model's *generation* ability; SASS uses their mean over $\mathcal{S}_{\text{gen}}$ as the generation-side term of the energy below.

Outcome-verified calibration pass. On N_{cal} calibration samples drawn from the training set, the warmup model greedily regenerates answers, and a lightweight *task-specific outcome verifier*[1] labels each answer as correct or incorrect, yielding correctness labels $\mathbf{y}_{\text{corr}} \in \{0,1\}^{N_{\text{cal}}}$. For every candidate direction $j \in \mathcal{P}$, we cache the answer-masked right projection $P_{s,j} = \langle \bar{\mathbf{h}}_s, \mathbf{v}_j \rangle$ of sample s, with $M_{s,t} = \mathbf{1}\{y_{s,t} \neq -100\}$ and $\bar{\mathbf{h}}_s = \sum_t M_{s,t}\mathbf{h}_{s,t} / \sum_t M_{s,t}$, forming $\mathbf{P} \in \mathbb{R}^{N_{\text{cal}} \times |\mathcal{P}|}$. This single training-time pass requires no extra trainable verifier and reuses gold labels. The design is inspired by outcome-verified RL (Liu et al., 2025; Kadavath et al., 2022).

Outcome-verified energy. For a candidate configuration $z = (\mathcal{S}_{\text{gen}}, \mathcal{S}_{\text{ver}})$, the per-sample verification norm is $\rho_s = \sqrt{\sum_{j \in \mathcal{S}_{\text{ver}}} P_{s,j}^2}$. Let $\mathcal{C} = \{s : y_{\text{corr},s} = 1\}$ and $\bar{\mathcal{C}} = \{s : y_{\text{corr},s} = 0\}$. The two group means are

$$\mu_c = \frac{1}{|\mathcal{C}|} \sum_{s \in \mathcal{C}} \rho_s, \qquad \mu_{\bar{c}} = \frac{1}{|\bar{\mathcal{C}}|} \sum_{s \in \bar{\mathcal{C}}} \rho_s, \tag{2}$$

and $\sigma_c, \sigma_{\bar{c}}$ are the corresponding sample standard deviations. We then measure how well the TSVD subspace separates the two groups by Cohen's d and the point-biserial correlation:

$$d_{\text{ver}} = \frac{|\mu_{\bar{c}} - \mu_c|}{\sqrt{(\sigma_c^2 + \sigma_{\bar{c}}^2)/2}}, \qquad r_{\text{pb}} = \text{corr}\big(\{\rho_s\}, \mathbf{y}_{\text{corr}}\big). \tag{3}$$

The energy combines TSVD discrimination with TSGD generation alignment:

$$E(z) = -\,w_1\, d_{\text{ver}} \;-\; w_{1b}\,|r_{\text{pb}}| \;-\; w_2\,\overline{\text{AMS}}(\mathcal{S}_{\text{gen}}), \tag{4}$$

where $\overline{\text{AMS}}(\mathcal{S}_{\text{gen}}) = \frac{1}{k_g} \sum_{i \in \mathcal{S}_{\text{gen}}} \text{AMS}_i$ and $(w_1, w_{1b}, w_2) = (1.0, 0.5, 0.3)$ by default. The outcome labels make correct/incorrect separation an explicit objective: d_{ver} cheaply measures the standardized separation induced by a candidate TSVD subset. Verification relevance is therefore supervision-induced, not an intrinsic property guaranteed by the pretrained spectrum.

Simulated-annealing search. We minimize $E(z)$ with a Metropolis sampler (Kirkpatrick et al., 1983; Metropolis et al., 1953; Hastings, 1970) under geometric cooling $T_t = T_0\,\alpha_{\text{SA}}^t$. Each step swaps one direction of $\mathcal{S}_{\text{gen}}$ or $\mathcal{S}_{\text{ver}}$ for an unused candidate, or, with small probability, exchanges the two roles; the proposal is accepted with probability $\min\big(1, \exp(-\Delta E / T_t)\big)$. Incoming directions stay outside $\mathcal{S}_{\text{gen}} \cup \mathcal{S}_{\text{ver}}$, so the subsets remain disjoint, and because ρ_s changes by one term per swap each step costs $O(N_{\text{cal}})$. We keep the lowest-energy $(\mathcal{S}_{\text{gen}}, \mathcal{S}_{\text{ver}})$ across restarts. Figure 2 illustrates the calibration pass and Algorithm 1 (§B.5) gives the full procedure.

2.3 THREE-PHASE TRAINING PIPELINE

Phase 1: Warmup (steps 1 to T_w). Initialize $R(0)$ from the bottom-r singular triplets and set $\mathbf{W}_{\text{base}}^{(1)} = \mathbf{W}_0 - sR(0)$, so the initial implemented weight equals $\mathbf{W}_0$. Train all three residual factors with $\mathcal{L}_{\text{ortho}}^{\text{pre}}$ (§2.4, §B.7) and the unweighted cross-entropy $\mathcal{L}_{\text{warmup}} = \frac{1}{|\mathcal{B}|} \sum_j \mathcal{L}_{\text{CE}}(x_j)$. No projection constrains $R(t)$ to the initial bottom-r subspace after initialization.

[1] Choice-letter matching for commonsense, boxed-answer equivalence for math, alias matching for factual QA, and a normalized-similarity heuristic for code.

Phase 2: Direction Extraction (at step T_w). A one-time outcome-verified calibration pass computes AMS and the cached projections $\mathbf{P}$ with correctness labels $\mathbf{y}_{\text{corr}}$, after which simulated annealing selects $\mathcal{S}_{\text{gen}}$ and $\mathcal{S}_{\text{ver}}$ (§2.2; see Algorithm 1). Scalar coefficients are initialized to the pretrained singular values: $\alpha_i^{(0)} = \sigma_i$, $\beta_j^{(0)} = \sigma_j$. At extraction the implementation applies the scaled correction

$$\mathbf{W}_{\text{base}}^{(2)} = \mathbf{W}_{\text{base}}^{(1)} - s \sum_{i \in \mathcal{S}_{\text{gen}}} \sigma_i \mathbf{u}_i \mathbf{v}_i^\top - s \sum_{j \in \mathcal{S}_{\text{ver}}} \sigma_j \mathbf{u}_j \mathbf{v}_j^\top .$$

Since $\alpha_i(T_w^+) = \sigma_i$ and $\beta_j(T_w^+) = \sigma_j$, the selected-direction terms cancel this correction and $\mathbf{W}(T_w^+) = \mathbf{W}(T_w^-)$ (Theorem 1).

Phase 3: Full Training (steps $T_w + 1$ to T). Jointly train the residual adapter (continued from warmup, not reinitialized), the TSGD scalars $\{\alpha_i\}$, and the TSVD scalars $\{\beta_j\}$, using the adaptive loss in §2.4. The direction sets $\mathcal{S}_{\text{gen}}, \mathcal{S}_{\text{ver}}$ remain fixed for the entire phase.

2.4 Adaptive Loss with Gen–Ver Coordinate Reweighting

Orthogonality loss. We add an orthogonality term that changes at extraction; before extraction it follows SORSA (Cao & Song, 2025) and DisLoRA (Yifei et al., 2025): $\mathcal{L}_{\text{ortho}}^{\text{pre}} = \frac{1}{2}(\|U_{\text{res}}^\top U_{\text{res}} - I_r\|_F + \|V_{\text{res}}^\top V_{\text{res}} - I_r\|_F)$. After extraction it constrains only the residual right singular subspace, with $V_{\text{gen}} = [\mathbf{v}_i]_{i \in \mathcal{S}_{\text{gen}}} \in \mathbb{R}^{n \times k_g}$ and $V_{\text{ver}} = [\mathbf{v}_j]_{j \in \mathcal{S}_{\text{ver}}} \in \mathbb{R}^{n \times k_v}$:

$$\mathcal{L}_{\text{ortho}} = \frac{\|V_{\text{res}}^\top V_{\text{gen}}\|_F^2 + \|V_{\text{res}}^\top V_{\text{ver}}\|_F^2}{k_g + k_v}, \tag{5}$$

averaged over participating directions and target matrices; it does not constrain the left residual factor or keep $R(t)$ in its initialization subspace.

Gen–Ver Spectral Coordinate Reweighting (GVSCR). We use the deployed $k_g = k_v$. At a probe layer ℓ, $\mathbf{h}_{i,t}^{(\ell)}$ is the hidden state, $M_{i,t} = \mathbf{1}\{y_{i,t} \neq -100\}$ is the answer-span mask, and $\mathbf{V}_g^{(\ell)} \in \mathbb{R}^{k_g \times n}$ and $\mathbf{V}_v^{(\ell)} \in \mathbb{R}^{k_v \times n}$ are the selected TSGD and TSVD right directions (row-stored; $\mathbf{V}_g = V_{\text{gen}}^\top$, $\mathbf{V}_v = V_{\text{ver}}^\top$). The reweighting is then

$$\begin{aligned}
\mathbf{g}_i^{(\ell)} &= \frac{\sum_t M_{i,t} \big(\mathbf{h}_{i,t}^{(\ell)} \mathbf{V}_g^{(\ell)\top}\big)}{\sum_t M_{i,t}}, \\
\mathbf{v}_i^{(\ell)} &= \frac{\sum_t M_{i,t} \big(\mathbf{h}_{i,t}^{(\ell)} \mathbf{V}_v^{(\ell)\top}\big) \odot |\boldsymbol{\beta}^{(\ell)}|}{\sum_t M_{i,t}}, \\
\mathbf{g}_i &= \text{norm}_2\big(\text{concat}_\ell\, \mathbf{g}_i^{(\ell)}\big), \quad \mathbf{v}_i = \text{norm}_2\big(\text{concat}_\ell\, \mathbf{v}_i^{(\ell)}\big), \\
s_i &= \langle \mathbf{g}_i, \mathbf{v}_i \rangle \in [-1, 1], \quad \bar{s} := |\mathcal{B}|^{-1} \sum_i s_i,
\end{aligned} \tag{6}$$

$$\begin{aligned}
\mu_t &= m\mu_{t-1} + (1-m)\bar{s}, \\
\text{var}_t &= m\,\text{var}_{t-1} + (1-m)\overline{(s_i - \mu_t)^2}, \\
c_i^{\text{coord}} &= \text{sigmoid}\left(\frac{s_i - \mu_t}{\sqrt{\text{var}_t + \epsilon}}\right),
\end{aligned} \tag{7}$$

$$\begin{aligned}
\widetilde{c}_i &= \begin{cases} c_i^{\text{coord}}(1 - H_i)^p, & \text{entropy gate enabled,} \\ c_i^{\text{coord}}, & \text{otherwise,} \end{cases} \\
\tau &= (t_3 - T_{\text{EMA}})_+, \qquad \lambda_{\text{eff}}(t) = \lambda_{\text{conf}} \min\left(1, \frac{\tau}{T_\lambda}\right), \\
w_i &= \begin{cases} 1, & t_3 \le T_{\text{EMA}}, \\ 1 + \lambda_{\text{eff}}(1 - \widetilde{c}_i)^\gamma, & t_3 > T_{\text{EMA}}, \end{cases} \quad \mathcal{L}_{\text{task}} = \frac{\sum_i w_i \mathcal{L}_{\text{CE}}(x_i)}{\sum_j w_j}.
\end{aligned} \tag{8}$$

EMA moments are averaged across ranks when local batch sizes match; H_i is normalized mean token entropy on the answer span (Kadavath et al., 2022); and t_3 is the Phase-3 step count. Defaults

are $\epsilon = 10^{-8}$, $T_{\text{EMA}} = 50$, $T_\lambda = 100$, entropy gate enabled with $p = 1$, $\lambda_{\text{conf}} = 0.5$, $\gamma = 2$, and $m = 0.9$. The detached c_i^{coord} is coordinate-dependent, not a basis-invariant or calibrated confidence probability. Algorithm 2 and theorem 6 give the procedure and weight properties.

Finally, $\mathcal{L}_{\text{task}}$ and $\mathcal{L}_{\text{ortho}}$ are combined with the EMA-smoothed mean-normalization scheme of DisLoRA (Yifei et al., 2025), which can automatically tune the α_{task} and α_{ortho} (formula reproduced in Eq. (A3), §B.7).

3 Experiments

We evaluate SVDoRA on commonsense reasoning, mathematical reasoning, code generation, and factual question answering. We also train linear probes to test whether the selected TSVD features decode answer correctness.

3.1 Experimental Setup

Configuration and comparison sources. Table A16 reports the settings used in our runs. Daggered Qwen2.5-7B commonsense rows are taken from Yifei et al. (2025, Table 2). SVDoRA's method-specific hyperparameters are selected once on commonsense 1.5B and reused for the remaining experiments. At rank 16 this adds 24 trainable scalars per target matrix (Table A15).

Commonsense reasoning. We train on `commonsense_170k` and evaluate zero-shot on eight tasks: BoolQ (Clark et al., 2019), PIQA (Bisk et al., 2020), Social IQA (Sap et al., 2019), HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2021), ARC-Easy and ARC-Challenge (Clark et al., 2018), and OpenBookQA (Mihaylov et al., 2018). This is the only family run at both model scales (1.5B and 7B) and across three ranks $r \in \{8, 16, 32\}$, against six baselines (LoRA, PiSSA, MiLoRA, SORSA, LoRA-DASH, and DisLoRA); the outcome verifier is choice-letter matching.

Mathematical reasoning. We train on `metamath_40k`, a 40k subset of MetaMathQA (Yu et al., 2024) that rewrites GSM8K (Cobbe et al., 2021) and MATH (Hendrycks et al., 2021) training problems into diverse chain-of-thought traces, and evaluate on GSM8K, MATH (levels 3–5), and AQuA-RAT (Ling et al., 2017); the verifier checks boxed-answer equivalence. This and the two families below run at r=16 on Qwen2.5-7B-Instruct and Llama-3-8B-Instruct, adding a transfer test across model families.

Code generation. We train on `magicoder_evol_110k`, extending the target modules to the FFN, and report execution-based pass@1 on HumanEval (Chen et al., 2021) and MBPP (Austin et al., 2021). During calibration the verifier is a normalized-similarity heuristic rather than execution: sandboxed unit tests are too slow for the one-shot pass, and environmental failures would corrupt the outcome labels (appendix H).

Factual question answering. We train on `triviaqa_50k` and report TriviaQA (Joshi et al., 2017) exact match plus TruthfulQA MC1 (Lin et al., 2022) evaluated zero-shot on the same adapter; the verifier is alias matching against the gold answer set.

3.2 Commonsense Reasoning

Table 1 reports the complete 7B task breakdown at $r \in \{8, 16, 32\}$. SVDoRA's displayed 7B averages are highest at all three ranks (88.4, 88.1, and 87.5%); at 1.5B it is highest at $r = 8, 16$ but trails LoRA-DASH by 0.2 pp at $r = 32$ (appendix C). These are displayed score comparisons.

3.3 Mathematical Reasoning

Table 2 shows that the SVDoRA hyperparameters selected on commonsense also transfer to a new task family and to a different model family. On Llama-3-8B-Instruct SVDoRA is the best method on the GSM8K+MATH average (48.4% versus 48.1% for DisLoRA), matching DisLoRA on GSM8K (76.8%) and leading on MATH (19.9% versus 19.4%). On Qwen2.5-7B-Instruct it places second

Table 1: Commonsense accuracy (%) on Qwen2.5-7B-Instruct across $r \in \{8, 16, 32\}$. Non-daggered rows are our runs with 3 seeds; daggered rows are taken from Yifei et al. (2025, Table 2). **Bold**: best within each rank; underline: second best.

Method	Rank	Param. (%)	BoolQ	PIQA	SIQA	HellaS.	WinoG.	ARC-c	ARC-e	OBQA	Avg.
LoRA†	8	0.07	70.5±2.7	84.9±1.3	79.4±0.8	73.7±3.1	85.0±1.2	86.0±0.9	93.6±0.7	88.4±1.6	82.7±1.5
PiSSA	8	0.07	71.7±0.4	89.4±0.1	80.1±0.7	94.3±0.2	84.3±0.9	86.3±0.5	94.9±0.3	90.0±1.4	86.4±0.4
MiLoRA†	8	0.07	69.3±2.1	81.0±1.9	79.6±1.3	87.1±1.8	84.7±1.4	87.8±1.2	94.6±0.8	90.2±1.5	84.3±2.7
SORSA†	8	0.07	68.6±2.2	85.0±1.4	78.5±0.9	90.9±0.9	84.6±1.3	86.3±1.0	94.4±0.6	86.4±1.3	84.3±1.6
LoRA-DASH	8	0.07	72.5±0.4	89.6±0.2	80.1±0.4	94.6±0.1	84.1±0.2	87.5±0.9	95.5±0.0	90.7±0.3	86.8±0.2
DisLoRA†	8	0.07	72.3±0.4	85.5±0.6	79.3±0.8	92.3±0.7	83.7±0.6	87.3±0.7	95.1±0.7	88.6±0.4	85.5±1.3
SVDoRA (ours)	8	0.07	**74.3±0.3**	**90.1±0.3**	**81.7±0.3**	**95.0±0.0**	**87.9±0.6**	**89.4±0.5**	**96.1±0.1**	**92.3±0.6**	**88.4±0.0**
LoRA†	16	0.13	71.4±2.4	77.1±1.9	78.9±0.9	88.3±2.7	83.3±1.1	85.2±0.8	93.3±0.6	86.6±1.7	83.0±1.3
PiSSA	16	0.13	72.2±0.6	88.8±0.1	80.1±0.7	94.1±0.0	84.6±0.5	84.8±0.4	94.0±0.4	88.1±1.5	85.8±0.2
MiLoRA†	16	0.13	69.2±2.0	85.5±1.6	74.9±1.8	92.3±1.1	85.5±1.3	86.7±1.3	94.1±0.7	89.2±1.4	84.7±3.0
SORSA†	16	0.13	71.4±1.7	85.1±1.8	79.6±0.7	91.2±0.7	85.4±0.9	84.8±1.1	94.7±0.5	90.4±0.4	85.3±1.5
LoRA-DASH	16	0.13	73.1±0.4	89.2±0.3	80.3±0.5	**94.7±0.0**	85.1±0.5	86.9±0.5	95.0±0.4	89.7±0.5	86.7±0.1
DisLoRA†	16	0.13	72.7±1.3	85.9±0.4	79.2±0.7	92.8±0.8	85.4±0.5	87.0±0.9	95.0±0.3	90.4±0.4	86.1±1.0
SVDoRA (ours)	16	0.13	**74.0±0.3**	**90.0±0.4**	**81.1±0.1**	94.7±0.0	**88.6±0.6**	**88.5±0.3**	**96.1±0.0**	**91.7±1.6**	**88.1±0.4**
LoRA†	32	0.26	71.2±2.3	80.3±1.9	78.6±0.8	85.1±2.6	84.5±0.9	84.7±0.8	93.8±0.5	89.0±1.5	83.5±1.2
PiSSA	32	0.26	72.4±0.4	88.4±0.3	79.2±0.9	93.3±0.1	83.5±0.2	83.2±0.9	93.3±0.1	86.0±0.3	84.9±0.2
MiLoRA†	32	0.26	72.0±1.9	83.8±1.7	78.9±1.2	92.7±1.0	86.1±1.5	85.1±1.4	94.1±0.6	90.8±1.1	85.4±2.6
SORSA†	32	0.26	70.3±1.6	85.1±1.7	78.2±0.5	92.7±0.6	84.9±0.7	85.9±0.9	94.6±0.4	90.9±0.5	85.3±1.4
LoRA-DASH	32	0.26	73.1±0.4	88.5±0.3	79.6±0.5	**94.7±0.1**	85.2±1.1	85.9±1.0	94.2±0.3	89.7±0.6	86.4±0.1
DisLoRA†	32	0.26	73.3±1.1	85.7±1.4	80.0±0.7	93.1±0.5	85.0±0.4	87.5±1.3	95.1±0.6	89.8±0.5	86.2±1.4
SVDoRA (ours)	32	0.26	**73.9±0.8**	**89.2±0.8**	**81.2±0.3**	94.4±0.4	**87.2±0.6**	**87.9±0.5**	**95.2±0.1**	**91.2±0.6**	**87.5±0.4**

Table 2: Mathematical reasoning accuracy (%) at rank r=16, mean±std over 3 seeds. GSM8K+MATH averages the two in-distribution tests covered by MetaMathQA; AQuA-RAT is a 254-item out-of-distribution multiple-choice probe and is included in the three-task Avg. **Bold**: best per model on the Avg. and GSM8K+MATH columns; underline: second best.

Method	GSM8K	MATH	AQuA	Avg.	GSM8K+MATH
Qwen2.5-7B-Instruct					
LoRA	83.7±0.4	37.6±0.3	41.1±2.6	54.1±0.9	60.6±0.0
PiSSA	82.7±0.4	36.6±0.4	36.6±0.0	52.0±0.3	59.7±0.4
MiLoRA	83.5±0.1	37.7±0.1	36.6±1.2	52.6±0.5	60.6±0.1
DisLoRA	84.5±0.0	38.9±0.4	39.2±2.6	**54.2±1.0**	**61.7±0.2**
SVDoRA	83.8±0.6	37.8±0.0	34.6±2.8	52.1±1.1	60.8±0.3
Llama-3-8B-Instruct					
LoRA	76.5±0.0	17.9±0.3	19.7±3.9	38.0±1.2	47.2±0.1
PiSSA	73.4±0.6	16.1±0.0	15.2±0.2	34.9±0.1	44.7±0.3
MiLoRA	74.6±0.3	17.1±0.4	18.7±0.6	36.8±0.2	45.9±0.0
DisLoRA	76.8±0.3	19.4±0.1	23.4±0.2	**39.9±0.0**	48.1±0.1
SVDoRA	76.8±0.3	19.9±0.2	18.3±0.2	38.4±0.1	**48.4±0.1**

(60.8% versus 61.7%), 0.2 pp ahead of LoRA and MiLoRA. SVDoRA is below DisLoRA on AQuA-RAT (Qwen 34.6±2.8 vs. 39.2±2.6; Llama 18.3±0.2 vs. 23.4±0.2) and on the three-task average (52.1 vs. 54.2; 38.4 vs. 39.9). The Qwen AQuA intervals overlap.

3.4 Code Generation

Code is the family with the least consistent outcome across base models. On Llama-3-8B-Instruct SVDoRA has the best average among the four completed methods (60.3% versus 57.8% for DisLoRA), including the highest HumanEval score (67.9%); on Qwen2.5-7B-Instruct it places fourth of five (65.5% versus 68.5%), with the gap concentrated in HumanEval. This pattern is consistent with a limitation of similarity-based calibration, but task family and verifier type are confounded (appendix H).

Table 3: Code generation pass@1 (%) at rank r=16, mean±std over 2 seeds; entries without ± are single-seed runs and "-" marks configurations not completed within the compute budget. **Bold**: best HumanEval+MBPP average per model; underline: second best.

Method	Qwen2.5-7B-Instruct			Llama-3-8B-Instruct		
	HumanEval	MBPP	Avg.	HumanEval	MBPP	Avg.
LoRA	76.7±0.2	59.6±0.6	68.1±0.2	64.2±1.3	49.3±1.0	56.7±1.2
PiSSA	70.9±0.7	54.3±0.4	62.6±0.5	-	-	-
MiLoRA	74.5±1.4	61.0±0.5	67.7±0.4	64.4±1.1	50.8±0.2	57.6±0.5
DisLoRA	75.2	61.8	**68.5**	62.2±0.8	53.4±0.1	57.8±0.3
SVDoRA	70.9	60.1	65.5	67.9	52.7	**60.3**

Table 4: Factual QA accuracy (%) at rank r=16, mean±std over 3 seeds. TruthfulQA MC1 is evaluated zero-shot on the same TriviaQA-trained adapter. **Bold**: best two-task average per model; underline: second best.

	Qwen2.5-7B-Instruct			Llama-3-8B-Instruct		
Method	TriviaQA	TruthfulQA	Avg.	TriviaQA	TruthfulQA	Avg.
LoRA	58.9±0.2	27.8±0.3	43.3±0.2	72.3±0.2	24.7±0.2	48.5±0.2
PiSSA	58.4±0.1	25.9±1.0	42.1±0.5	70.3±0.3	22.7±0.4	46.5±0.3
MiLoRA	59.4±0.1	28.0±1.5	43.7±0.7	72.3±0.3	23.8±1.0	48.0±0.5
DisLoRA	**59.9**±0.1	**30.8**±0.3	**45.4±0.1**	**74.6**±0.1	25.3±0.6	**49.9±0.3**
SVDoRA	59.5±0.3	29.7±0.8	44.6±0.3	74.3±0.1	**25.6**±1.0	49.9±0.5

3.5 Factual Question Answering

On factual QA (Table 4), SVDoRA is second on Qwen2.5-7B-Instruct (44.6% vs. 45.4%); on Llama-3-8B-Instruct both averages read 49.9%, with DisLoRA bolded. The TruthfulQA column is the more informative one: it is a zero-shot transfer evaluation of an adapter trained only on TriviaQA, and SVDoRA is the best method on Llama-3-8B (25.6%) while remaining within 1.1 pp of the best on Qwen.

3.6 Do TSVD Directions Carry a Verification Signal?

We test whether the TSVD features selected by SVDoRA linearly decode answer correctness.

Configuration. At decoder layer L21 we extract the last non-EOS generated-token state h and form each subspace by concatenating its k projections $\langle h, v_i \rangle$ onto the selected right-singular directions of the q/k/v/o modules ($d_{\text{probe}} = 4k$); a supervised $\text{Linear}(d_{\text{probe}}, 2)$ classifier predicts correctness over five stratified 80/20 splits with train-fold whitening. Beyond TSVD and TSGD we include six controls: the minor-σ residual band, the near-null space, random real singular directions, coordinate-permuted TSVD directions, a sparse Johnson–Lindenstrauss projection, and a Gaussian orthonormal basis; the raw hidden state is probed without projection.

Result. At $d_{\text{probe}} = 16$, matching the deployed k_g=k_v=4, TSVD attains the highest math accuracy 0.635 ± 0.042 and weighted F1 (0.639) among the eight subspaces (majority-class accuracy 0.624, majority-class weighted F1 0.479; AUROC 0.672). On accuracy and weighted F1 it exceeds TSGD, the minor-σ residual band, and the random-feature controls (Rand-spectrum, Sparse-rand, and Random), while its AUROC only ties the best control (null-space). Even so, the comparison is non-exhaustive and does not show that the model detects or corrects its own errors (appendix I, Table A8, and Figure 3).

3.7 Ablations and Takeaways

On Qwen2.5-1.5B commonsense, outcome-verified annealing (SASS) is the strongest selector we compared (79.32% versus 76.97% random and 78.88% for the legacy correctness-agnostic second-

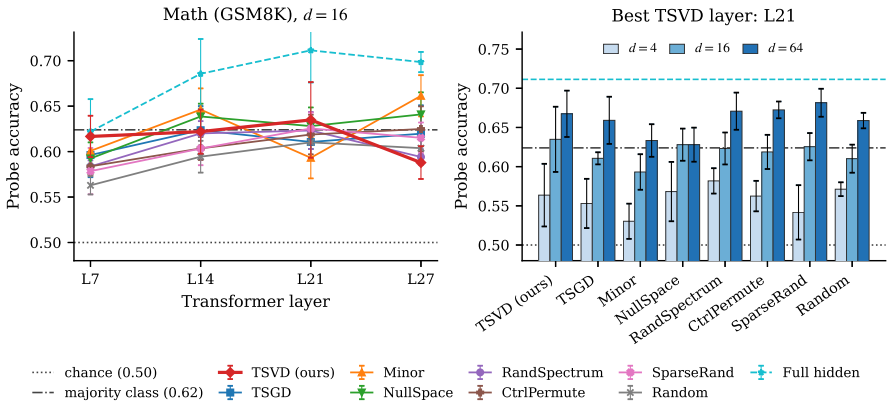

Figure 3: Probe accuracy on math (GSM8K), Qwen2.5-1.5B-Instruct. *Left:* layer-wise accuracy at $d_{\text{probe}} = 16$, matching the deployed $k_g{=}k_v{=}4$. *Right:* best-layer accuracy across the dimension grid. Labels are unbalanced, so the baseline is the majority rate (0.624), not 0.500.

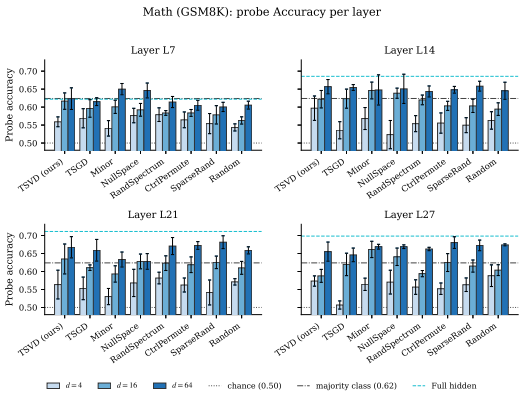

Figure 4: Probe accuracy per layer, math (GSM8K), Qwen2.5-1.5B-Instruct; majority-class floor 0.624.

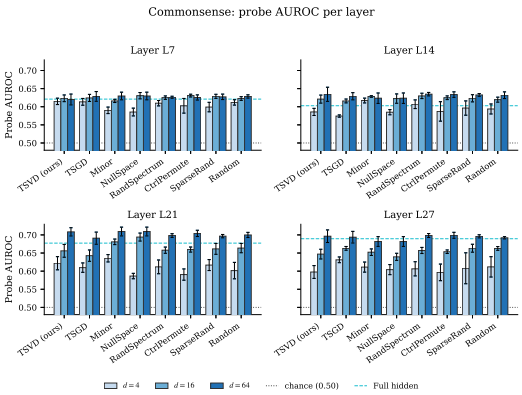

Figure 5: Probe AUROC per layer, commonsense, Qwen2.5-1.5B-Instruct, with chance at 0.500.

order gradient top-k ranking (Variance-based Uncertainty Score, VUS); Table A3), and we also compared an online version of SASS, which is defeated by its offline version (Table A2), so we locked the SASS in the shared SVDoRA pipeline. The gap does not mean online SASS has no headroom: the present energy $E(z)$ and geometric Metropolis cooling were designed for a one-shot search, and may be a poor fit for repeated online reselection, and SASS can also be limited by how stably the Cohen's d in the energy function can be estimated. For Phase 3, which computes the loss by reweighting, whether it uses uniform CE, last-layer TSVD-norm reweighting, or GVSCR, the eight-task mean moves by at most 0.3 pp (Table A4), so the accuracy gain is carried by the selected directions rather than by the loss weight, with GVSCR occupying a narrow non-harmful region at $(\lambda_{\text{conf}}, \gamma) = (0.5, 2)$ (Table A6). The calibration size can also affect accuracy: cutting N_{cal} from 512 to 128 costs 0.43 pp, while raising it to 1024 yields no further gain (Table A5).

4 Conclusion

SVDoRA labels generated calibration answers against gold outcomes to select fixed generation-aligned (TSGD) and verification-relevant (TSVD) spectral directions after a one-time annealing pass. GVSCR reweights samples by an EMA-standardized TSGD–TSVD cosine. Commonsense is highest at $r{=}16$ on Qwen2.5-1.5B and 7B. The hyperparameters selected on commonsense show competitive transfer to factual QA and code generation.

AI USE STATEMENT

In this work, we used generative AI tools to implement methods (building distributed training pipeline, and analysis code) and to assist in writing proofs and in drafting theoretical statements that appear in the appendix. We have not used generative AI tools to generate synthetic data sets, assist with translation, clean or reformat a dataset, or support qualitative and thematic data analysis, and those tasks are not applicable to this work. We also did not use generative AI tools to propose the research hypotheses but only to refine them; those core ideas were proposed by the authors. Additionally, we used generative AI tools to draft and edit the manuscript for readability, search and summarize existing literature, and format references. We did not use generative AI tools to create the main body of scientific figures but only to create some icons in them. AI-assisted proofs were checked line by line by the authors. AI-assisted code was executed and compared against the algorithms and claims in the paper. AI-suggested citations were verified against the cited sources. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide the details needed to reproduce SVDoRA in the main text and appendix rather than in this paragraph. The four-way partition, outcome-verified energy, and training losses are specified in §2; §3.1 reports the recipe for our runs. Pseudocode for outcome-guided spectral annealing and GVSCR is given in Algorithms 1 and 2. Assumptions, formal claims, and proofs are collected in §B. Hyperparameters, hardware, and seed protocols for our main-experiment and ablation runs are listed in Tables A16 and A18. Public pretrained checkpoints and standard training and evaluation splits are used throughout. Anonymized source code for the method, training, and evaluation will be provided in the supplementary zip archive together with the appendix, following the ICLR Author Guidelines.

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. *arXiv preprint arXiv:2108.07732*, 2021.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 7432–7439, 2020.

Yang Cao and Zhao Song. SORSA: Singular values and orthonormal regularized singular vectors adaptation of large language models. *arXiv preprint arXiv:2409.00055*, 2025.

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. *arXiv preprint arXiv:2107.03374*, 2021.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics: Human language technologies, volume 1 (long and short papers)*, pp. 2924–2936, 2019.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

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. *arXiv preprint arXiv:2110.14168*, 2021.

DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970. doi: 10.1093/biomet/57.1.97.

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 *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/hash/be83ab3ecd0db773eb2dc1b0a17836a1-Abstract-round2.html.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, 2017. URL https://aclanthology.org/P17-1147/.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220 (4598):671–680, 1983. doi: 10.1126/science.220.4598.671.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252, 2022. URL https://aclanthology.org/2022.acl-long.229/.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 158–167, 2017. URL https://aclanthology.org/P17-1015/.

Xiaoyuan Liu, Tian Liang, Zhiwei He, Jiahao Xu, Wenxuan Wang, Pinjia He, Zhaopeng Tu, Haitao Mi, and Dong Yu. Trust, but verify: A self-verification approach to reinforcement learning with verifiable rewards. *Advances in Neural Information Processing Systems*, 38:130475–130501, 2025.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. PiSSA: Principal singular values and singular vectors adaptation of large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 121038–121072, 2024. doi: 10.52202/079017-3846.

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953. doi: 10.1063/1.1699114.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 2381–2391, 2018.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025a.

Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025b.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. Social iqa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pp. 4463–4473, 2019.

Chongjie Si, Zhiyi Shi, Shifan Zhang, Xiaokang Yang, Hanspeter Pfister, and Wei Shen. Unleashing the power of task-specific directions in parameter efficient fine-tuning. In *International Conference on Learning Representations*, volume 2025, pp. 78837–78869, 2025.

Hanqing Wang, Yixia Li, Shuo Wang, Guanhua Chen, and Yun Chen. MiLoRA: Harnessing minor singular components for parameter-efficient LLM finetuning. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 4823–4836, 2025. doi: 10.18653/v1/2025.naacl-long.248. URL https://aclanthology.org/2025.naacl-long.248/.

Yixuan Weng, Minjun Zhu, Fei Xia, Bin Li, Shizhu He, Shengping Liu, Bin Sun, Kang Liu, and Jun Zhao. Large language models are better reasoners with self-verification. In *Findings of the Association for Computational Linguistics: EMNLP*, pp. 2550–2575, 2023.

She Yifei, Xinhao Wei, and Yulong Wang. DisLoRA: Task-specific low-rank adaptation via orthogonal basis from singular value decomposition. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 13740–13755, Suzhou, China, nov 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.694. URL https://aclanthology.org/2025.emnlp-main.694/.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. MetaMath: Bootstrap your own mathematical questions for large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, 2019.

Fuxiang Zhang, Jiacheng Xu, Chaojie Wang, Ce Cui, Yang Liu, and Bo An. Incentivizing LLMs to self-verify their answers. *Advances in Neural Information Processing Systems*, 38:4800–4829, 2025.