Param-Mix · data-mixture interpolation

Olivia

1 · PROBLEM

what and why

Optimizing data mixtures for large language models remains computationally prohibitive because identifying the best mixture ratios typically requires repeated full-scale training runs guided by heuristic tuning. Existing methods are primarily optimized for language modeling loss such as perplexity, but reductions in perplexity do not reliably translate to improved downstream reasoning, coding, and mathematical performance. This project investigates whether parameter-space interpolation can serve as a high-fidelity proxy for mixed-data optimization by exploiting the shared first-order optimization dynamics between independently trained domain experts and mixed-domain training trajectories.

2 · APPROACH

model / method, and why this one

We propose a framework that approximates mixed-data training using linear interpolation between independently trained domain-specific expert models. Given interpolation coefficients $\alpha \in \Delta K\text{-}1$, the merged parameters are constructed as: $\Theta merge(\alpha) = \Theta base + \Sigma k\ \alpha k(\Theta k - \Theta base)$ This formulation allows efficient exploration of the mixture simplex without retraining full models. The method is motivated by a second-order Taylor expansion analysis showing that mixed-data optimization and parameter interpolation share identical first-order gradient terms, while discrepancies are confined to second-order curvature interactions involving Hessian-gradient products Hkgj. We additionally fit LightGBM regressors to model the performance surface across downstream benchmark domains and optimize a utility function over predicted benchmark scores.

3 · DATASET

source, size, preprocessing

We use a large-scale mid-training corpus partitioned into four primary domains: mathematics, code, supervised fine-tuning (SFT) data, and web/other data. The benchmark suite spans reasoning, mathematical problem solving, coding, language understanding, and professional knowledge tasks, including GSM8K, MATH, HumanEval, MBPP, GPQA, MMLU, ARC, BBH, and TriviaQA. Preprocessing • Domain clustering and token filtering to enforce structural purity. • Hierarchical decomposition into fine-grained sub-clusters (e.g., formal mathematics, synthetic study notes, code repositories, long chain-of-thought supervision, and academic papers). • Normalization of text distributions across all training segments.

4 · METRICS

how you'll know it worked

Category / Metric Target / Interpretation Experiment Proxy Validation Spearman rank correlation coefficient Target $\rho > 0.9$ between predicted interpolation performance (ρ) and full training trajectories Downstream Average benchmark accuracy Evaluated across 5 key capability domains: math, code, Capability knowledge, language, and reasoning Interference Hessian interaction matrices & task- Quantify cross-domain interference and optimization Analysis vector cosine similarities orthogonality to identify breakdown points

5 · COMPUTE

training time, memory, dependencies

The project requires approximately 2x AMD Instinct MI300X GPU accelerators with a combined memory budget near 384GB HBM memory to support distributed sparse Mixture-of-Experts mid-training experiments. Estimated Wall- Phase Hardware Notes / Scope Clock Domain-Specific Expert 2x AMD Instinct ~7-9 days Multiple 5B-25B token expert training runs with Training MI300X shared initialization Interpolation Sweeps & 2x AMD Instinct ~2-3 days Exploring mixture simplex and training LightGBM Surface Fitting MI300X performance regressors Benchmark Evaluation & 1x AMD Instinct ~1-2 days Validation verification runs on sampled mixture Verification MI300X configurations Software Dependencies • PyTorch, DeepSpeed distributed framework, FlashAttention • Distributed checkpoint interpolation utilities and custom Hessian-vector product modules • LightGBM-based regression models for downstream capability prediction • ROCm 6.x + official PyTorch ROCm wheel environment

6 · DELIVERABLES

`model, paper, demo`

Trained domain-specialized expert model checkpoints and a final optimized mid-trained language model. • Comprehensive benchmark evaluation reports and detailed visualization maps of capability landscapes projected over the mixture simplex. • Empirical analysis of first-order versus second-order optimization behavior during interpolation-based search sweeps. • Reproducible scripts for interpolation search, utility optimization, and downstream benchmark testing.

- · A working framework for efficient data mixture optimization using parameter interpolation without full retraining loops. •

7 · RISK

`what breaks first, and plan B`

what breaks first, and plan B Risk Likelihood Mitigation Higher-order curvature interactions Medium Constrain expert training to short horizons with constant learning cause parameter interpolation to diverge rates, preserve shared initialization across all experts, and from mixed-data training empirically validate via Hessian interaction analysis. Domain-specific task vectors do not Medium Monitor cosine similarity measurements continuously and perform remain sufficiently orthogonal, reducing rapid full-training verification runs on sampled mixture boundaries. rank correlation