CAN CAPABILITY ENHANCEMENT BE LOCALIZED?

Anonymous authors
Paper under double-blind review

ABSTRACT

Transformer-based language models have shown strong capabilities across many tasks and domains. However, prior work shows that, for a given task, not all attention heads matter equally. Specifically, they suggest that many *redundant* heads can be safely removed with little-to-no effect, while a small set of *important* heads, when ablated, severely damages the model's performance on the given task. In this work, we investigate a complementary aspect, called *Capability Enhancement Localization* (CEL): *given a model and a task, does there exist a small set of heads whose removal improves the task performance?* We first investigate this question *empirically* across models from different families. Using a simple two-stage routine that screens candidate heads and evaluates their joint ablations, we identify sparse head combinations whose removal improves task-specific performance without further fine-tuning. Motivated by these observations, we ask *theoretically whether such a phenomenon can naturally arise from training dynamics, and under what conditions*. To this end, we construct a simplified multi-head softmax attention model and two tasks with conflicting features, and show that gradient descent from small random initialization provably concentrates the conflicting features into particular heads. We further show that ablating such heads improves performance on one task while harming the other, even though training converges to a global optimum of the joint objective. This establishes a possible training-induced theoretical explanation for CEL. [1]

1 INTRODUCTION

Large language models (LLMs) (Vaswani et al., 2017; Hoffmann et al., 2022) have demonstrated strong capabilities across diverse tasks, including language understanding, text generation, question answering, code generation, and mathematical reasoning (Brown et al., 2020; Raffel et al., 2020; Rae et al., 2021), among others. A notable feature of LLMs is that a single set of parameters supports many different tasks, which can often be specified through natural-language instructions or a few demonstrations without task-specific parameter updates (Brown et al., 2020; Wei et al., 2021). This breadth of functionality motivates a line of work studying how a shared network support diverse capabilities and how its internal components (e.g., attention heads/layers) contribute to LLM performance on individual tasks.

Early works on LLM pruning demonstrated that the attention heads exhibit substantial redundancy and contribute unequally across different tasks. For example, Clark et al. (2019b) observed that attention heads within the same BERT layers (Devlin et al., 2019) often exhibit similar attention patterns, leading to overlapping behavior. Moreover, Michel et al. (2019) and Voita et al. (2019) showed that many attention heads can be pruned after training without substantially affecting the overall LLM performance. Together, those findings emphasize that the full complement of attention heads is generally not required at inference time for preserving task capabilities, motivating a closer look at which heads are dispensable and which are critical for particular tasks.

Besides the evidence of redundancy, recent works have identified attention heads with specialized roles. Examples include induction mechanism corresponding to in-context learning (Olsson et al., 2022), information retrieval from long contexts (Wu et al., 2025), and processing operands and operators for arithmetic tasks (Zhang et al., 2024), to name a few. Other work identified distinct heads that favor internal knowledge encoded on model's weights or external context when these

[1] Codes are available in here.

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.

information sources conflict (Jin et al., 2024). These analyses revealed how specific heads support or route computations under various scenarios. More recently, broader capability-localization studies complement these findings. In particular, Huang et al. (2025) identified a small group of *neurons* associated with shared capabilities, while Bair et al. (2026) leveraged a cheap compressed sensing method (i.e., LASSO (Tibshirani, 1996)) to identify *task-critical attention heads* corresponding to separate capabilities. Ablating just a few of these task-critical heads can *dramatically degrade* the LLM performance on corresponding task.

The above findings provided evidence of *redundant* and *task-critical* attention heads. In this work, we complement the these findings by asking the following counter-intuitive question: *given an arbitrary task and an LLM, is there a sparse set of attention heads such that ablating them improves the performance of that LLM on such task, without any further fine-tuning?* For convenience, we term this question *Capability Enhancement Localization* (CEL).

We note that beneficial head removal has already been observed in some *specific* scenarios. Held & Yang (2023) identify heads that interfere with performance in multilingual models and show that pruning them can improve performance for a target language. Jin et al. (2024) prune heads involved in conflicts between a model's internal knowledge and external context. More recently, Nguyen et al. (2025) observe that pruning certain heads can improve mathematical reasoning and use different pruned configurations to generate diverse answers for best-of-N selection. In this work, we ask whether sparse beneficial head ablations can be found across a *broader range of tasks and models*, without presupposing a particular source of interference. More importantly, we ask *how successful training can produce a head whose removal improves a task*. We answer this theoretical question in a simplified attention model, where task conflict and learned feature allocation give rise to beneficial head ablation.

Contribution. In this work, we study the *Capability Enhancement Localization* (CEL) question, which asks whether there exists a sparse set of attention heads whose ablation improves task performance without further fine-tuning, given a task. Our contributions are:

1. In Section 3, we consider multiple model families and tasks configuration. Using a simple two-stage approach, we found evidence of the CEL phenomenon: given a task, ablating a sparse set of heads can significantly improve models' task-specific performance.
2. Motivated by empirical evidence of CEL, we seek a potential theoretical explanation for whether and how this phenomenon naturally arises in LLM training. To this end, in Section 4, we construct a simplified softmax attention models, trained on two tasks with a feature conflict. We then show in Theorem 4.2 that, under standard GD training, such conflicting features can be encoded into a specific attention head, leading to the *localization* phenomenon.
3. We then further show in Theorem 4.2 that ablating such a head can improve the model's performance on a single task while harming the other, even though training converges to an optimum of the joint objective. Together with the localization phenomenon, this theoretically establishes one possible mechanism for CEL.

2 Related Work

Attention-head functionality. Studies of Transformer components reveal both redundancy and functional specialization. Many attention heads can be removed with little loss in task performance (Michel et al., 2019; Voita et al., 2019), while others participate in identifiable computations, including induction and information retrieval (Olsson et al., 2022; Wu et al., 2025). Beyond individual head functions, localization studies investigate components associated with factual associations, task-specific skills, and capabilities shared across examples (Meng et al., 2022; Panigrahi et al., 2023; Huang et al., 2025). At the attention-head level, Bair et al. (2026) identify sparse sets of task-critical heads whose joint ablation substantially degrades target-task performance while largely preserving performance on unrelated tasks. Together, these findings motivate examining how sparse interventions affect task performance, including whether their effects can be beneficial. In this work, *we instead focus on the enhancement regime*: identifying small sets of attention heads whose joint removal improves task performance without further fine-tuning. Furthermore, we also seek a *theoretical explanation* for how this phenomenon can arise in standard LLM training.

Performance enhancement through head pruning. Prior work has found benefits from head removal in several settings. Held & Yang (2023) show that pruning heads associated with interference across languages can improve performance for a target language. Jin et al. (2024) study conflicts between internal knowledge and external context, and show that pruning selected heads can improve question answering. In mathematical reasoning, Nguyen et al. (2025) observe improvements from pruning particular heads and use different pruned configurations to produce diverse answers for best-of-N selection. In this work, we consider a more general setting by examining sparse beneficial interventions across a broader range of tasks and models, and provide a potential theoretical explanation of how this phenomenon can arise in LLM training dynamics, an aspect that was ignored by prior works.

Mechanistic interpretability and circuit discovery. Mechanistic interpretability studies how internal components implement model behavior. Wang et al. (2022) identify an attention-head circuit for indirect object identification, while Conmy et al. (2023) use activation patching to automate the discovery of computational subgraphs. Such localization need not predict the effect of changing a model: Hase et al. (2023) find that causal-tracing results do not identify the best MLP layers for factual editing. In our experiments, head ablation measures the effect of a specified intervention but does not reveal the computation performed by the removed heads. Our theory instead shows, in a simplified model, how training can produce a head whose removal benefits one task.

3 Empirical Motivation: Sparse Beneficial Head Ablation

In this section, we introduce our notation, define task-specific performance gain under head ablations, and formulate the *Capability Enhancement Localization* (CEL) problem as discovering a small head subsets whose removal improves task-specific performance of the considered model.

3.1 Head ablation and evaluation

Notation. We consider a transformer-based LLM M with L layers and H attention heads per layer, given $N = LH$ heads in total. We denote the head at layer $\ell \in \{1, \dots, L\}$ and position $i \in \{1, \dots, H\}$ by $h_{\ell,i}$, and the set of all attention heads by $\mathcal{H} = \{h_{\ell,i} : \ell = \{1, \dots, L\}, i = \{1, \dots, H\}\}$. Our intervention targets a *sparse* subset of attention heads $S \subsetneq \mathcal{H}$, with the number of elements $|S| \leq k$ for $k \ll N$. In this work, we typically choose $k = 3$.

Head ablation, performance gain, and beneficial head subset. Given a set $S \subsetneq \mathcal{H}$ of attention heads, let $M_{\setminus S}$ denote the model obtained by setting output heads in S to zero at every token position before the attention output projection. We note that *all remaining parameters remain unchanged, and no fine-tuning is performed*. For a task T with a *test* dataset D_{test}, we denote the performance of the model M under the task T as

$$\text{Acc}_{T,D_{\text{test}}}(M) = \frac{1}{|D_{\text{test}}|} \sum_{(x,y)\in D_{\text{test}}} \mathbf{1}[M \text{ answers } x \text{ correctly under the task scorer}]$$

Under the same inference and scoring protocol for the intact and ablated models, we denote the *empirical ablation gain* as $\Delta_{T,D_{\text{test}}}(S) = \text{Acc}_{T,D_{\textbf{test}}}(M_{\setminus S}) - \text{Acc}_{T,D}(M)$. We then call the sparse subset of attention heads S *beneficial* on the test set D_{test} if $\Delta_{T,D_{\text{test}}}(S) > 0$.

Objective. Given a model M, a task T with a test set D, and a head budget k, we use a *disjoint discovery set* D_{dist} (i.e., $D_{\text{dist}} \cap D_{\text{test}} = \emptyset$) from the task T to select a head subset through a search routine $\mathcal{A}$:

$$\widehat{S} = \mathcal{A}(M, T, A, k), \qquad 1 \leq |\widehat{S}| \leq k.$$

The goal of CEL is to identify subsets $\hat{S}$ whose joint ablation yields a large positive test gain $\Delta_{T,D_{\text{test}}}(\widehat{S})$, while using only discovery data D_{dist} to guide selection. The selected subset $\hat{S}$ is fixed before test evaluation. By using this set up, our hope is that the identified subset $\hat{S}$ truly improving the task-specific performance on task T, rather than only overfitting the discovery set D_{dist}.

3.2 Experimental Setup

Models and tasks. We evaluate three models of different families: Llama-3.2-3B-Instruct, Qwen2.5-3B-Instruct, and SmolLM3-3B-Instruct (Grattafiori et al., 2024; Qwen et al., 2025; Bak-

Algorithm 1 Two-stage intervention selection and held-out evaluation

Require: Model M with head set $\mathcal{H}$; task T; discovery set D_{dist}; test set D_{test}
Require: Candidate count $p = 10$; head budget $k = 3$
Ensure: Selected subset $\widehat{S}$ and its test-set gain
For $D \in \{D_{\text{dist}}, D_{\text{test}}\}$, use $\text{Acc}_{T,D}(M_{\setminus S})$ and $\Delta_{T,D}(S)$ as defined above
Let $d_{T,D}(S)$ count examples in D answered correctly by M but incorrectly by $M_{\setminus S}$
Rank subsets on D_{dist} by higher gain, lower damage, higher accuracy, fewer heads, then lexicographic head order
Stage 1: Screen individual heads
for $h \in \mathcal{H}$ **do**
    Evaluate $\{h\}$ on D_{dist}
end for
$P \leftarrow$ heads in the top p singleton interventions
Stage 2: Select a joint intervention
$\mathcal{C} \leftarrow \{S \subseteq P : 1 \leq |S| \leq k\}$
for $S \in \mathcal{C}$ with $|S| \geq 2$ **do**
    Evaluate S on D_{dist} ▷ Reuse singleton results
end for
$\widehat{S} \leftarrow$ highest-ranked subset in $\mathcal{C}$ on D_{dist}
Freeze $\widehat{S}$
Held-out evaluation
Evaluate $\widehat{S}$ on D_{test} with a matched intact-model run
return $\widehat{S}$, $\text{Acc}_{T,D_{\text{test}}}(M_{\setminus \widehat{S}})$, $\Delta_{T,D_{\text{test}}}(\widehat{S})$

ouch et al., 2025). The number of layers and heads of considered models can be found in Table 2. For the tasks, we consider three capabilities using various datasets: (1) math capabilities with Arithmetic (Brown et al., 2020), (2) code generation with MBPP (Austin et al., 2021), and (3) language skills via HellaSwag (Zellers et al., 2019) and BoolQ (Clark et al., 2019a).

Intervention selection. For each model–task pair, we use a two-stage procedure on the discovery set D_{dist}. We first score individual heads to form a small candidate pool, then evaluate joint ablations of subsets drawn from that pool and select one subset with the best discovery-set performance. We fix this subset before measuring its gain on the held-out test set D_{test}. The test examples play no role in scoring heads or choosing the intervention. Details of the procedure is demonstrated in Algorithm 1. For the dataset MBPP, we use the official test set as D_{test} and subsample 374 examples for the training set as D_{dist}. For the datasets HellaSwag and BoolQ, we use the official validation datasets as the test set, and subsample 2,000 examples from the official training sets as discovery set D_{dist}. For the dataset Arithmetic, since it does not have a separate test set with official labels, we select from its official validation split of 20,000 examples 2,000 examples for D_{dist} and another disjoint 1,000 examples as D_{test}. For convenience, we summarize the number of examples of discovery and test sets for each validation tasks in Table 3.

Hardware configuration. We ran the experiments as Slurm jobs, each allocated one NVIDIA L40S GPU, two CPUs, 24 GB of system memory, and with batch size of 8.

3.3 Results

Table 1 reports the held-out accuracy changes for the model–task pairs studied here. Each gain compares an ablated model with an intact model evaluated under the same task protocol. The reported interventions improve held-out accuracy across Arithmetic, BoolQ, HellaSwag, and MBPP. For example, the SmolLM3 Arithmetic intervention raises accuracy from 70.40% to 83.50%, a gain of 13.10 percentage points. The gains vary across model–task pairs, so the evidence supports beneficial sparse ablations in these tested settings, rather than a claim that ablation will help every model on every task.

Our two-stage search is *deliberately simple*. It searches small subsets within a pool selected using individual-head effects, so it need not find the best subset among all heads. Its outcome can also depend on the discovery set: for Qwen on MBPP, the discovery-rank-1 intervention gained 0.20 per-

Table 1: Held-out performance of head ablations. SmolLM3 is evaluated in non-thinking mode.

Model	Task	Intact acc.	Ablated acc.	Gain (pp)	Rel. error reduction
SmolLM3	Arithmetic	70.40%	83.50%	+13.10	44.26%
	BoolQ	82.57%	84.92%	+2.35	13.51%
	HellaSwag	75.75%	77.05%	+1.29	5.34%
	MBPP	50.40%	52.00%	+1.60	3.23%
Qwen2.5-3B-Instruct	Arithmetic	89.50%	90.90%	+1.40	13.33%
	BoolQ	80.21	82.35	+2.14	10.82%
	HellaSwag	75.02	76.66	+1.63	6.54%
	MBPP	60.40%	62.20	+1.80	4.55%
Llama-3.2-3B-Instruct	Arithmetic	68.00%	74.30%	+6.30	19.69%
	BoolQ	78.72%	81.25%	+2.54	11.93%
	HellaSwag	71.64%	73.78%	+2.14	7.55%
	MBPP	47.80	53.00%	+5.20	9.96%

centage points on the held-out set when selected using 200 discovery examples, compared with 1.80 points when selected using 374 examples. This comparison describes sensitivity to the discovery setup; it does not establish that using more examples consistently increases the gain.

These results establish the empirical phenomenon but do not explain why training can produce components whose removal improves a particular task. We next study this question in a simplified attention model. The theory provides one possible mechanism for beneficial head ablation; it does not identify the mechanism responsible for the gains in the language models above.

4 A Theoretical Explanation of the Localization Capability Enhancement Phenomenon

As we saw previously, there is strong evidence suggesting the existence of the CEL phenomenon. But why does it happen? In this section, we will investigate this question via a simple model with *diagonal weights* and *small initialization scale*, inspired by prior works (Boix-Adsera et al., 2023; Pesme & Flammarion, 2023; Berthier, 2023). We then show that the CEL phenomenon can potentially arise naturally in the training trajectory if there is initially some conflict between tasks.

4.1 Tasks, Attention Model, and Training

In this section, we will first describe our simple model within which we study the CEL problem.

Data and task. Let U, V, W be independent Rademacher variables which take the value in $\{-1, 1\}$ with equal chance. Fix real value $a, b, c > 0$, and set $t = \frac{a}{2}$. We define two equally weighted training task, where We define two equally weighted training tasks:

$$A: \quad X = (U, V, 0), \quad Y = bV, \quad \text{and} \quad B: \quad X = (U, 0, W), \quad Y = aU + cW.$$

In this construction, the important feature is coordinate 0. On task A, the variable U appears in the input, but does not appear in the output, meaning that the ideal predictor should ignore coordinate 0. In contrast, on task B, the same variable U is useful, and should receive coefficient a. Therefore, in this case, *the two tasks disagree about the desired use of the same input coordinate* (a simple scenario of disagreements between tasks).

The situation with other coordinates is simpler. Coordinate 1, with variable V, carries the useful signal for task A, while coordinate 2 carries the useful W-signal for task B. In this case, for simplicity, we assume that the observations are noiseless, and we study *population* squared risk rather than finite-sample estimation error (which is a proxy to evaluate the population risk, as we did in the experiments above). Especially, *this ensures that any beneficial ablation in the analysis comes from task conflict and training dynamics, rather than from overfitting to a finite sample*.

Attention architecture and input encoding. We next consider a simple one-layer Transformer-style attention model with three attention heads. In this case, our goal is to understand how training

allocates already available features across heads, rather than how feature themselves are learned, we use a fixed coordinate encoding as follows. Let $e_0, e_1, e_2 \in \mathbb{R}^3$ denote the standard basis vectors. For an feature input $x = (x_0, x_1, x_2) \in \{-1, 0, 1\}^3$, we associate coordinate j with the query $z_j(x) = \sqrt{3}x_j e_j$. This means that the original input x will be encoded as the series of tokens before feeding to the transformer

$$z(x) = \begin{pmatrix} \sqrt{3}x_0 & 0 & 0 \\ 0 & \sqrt{3}x_1 & 0 \\ 0 & 0 & \sqrt{3}x_2. \end{pmatrix}$$

Therefore, the three coordinates of x are represented by three fixed feature directions. This is a *deliberate implication*, as we want that the feature basis is supplied to the model, while how these features associated to the attention heads is learned during the training process.

Fixed memory bank . For every coordinate $j \in \{0, 1, 2\}$, we introduce a fixed pair of memory tokens $m_{j,+} = e_j, \quad m_{j,-} = -e_j$. The query $z_j(x)$ attends only to the corresponding pair $(m_{j,+}, m_{j,-})$, using a fixed attention masks shared by all three heads. This paired construction serves two purposes. First, it gives every head exactly the same access to *every* feature, so no feature-to-head assignment is deterministically imposed in advance. Second, the symmetric pair $(+e_j, -e_j)$ yields a particular signed softmax response (which we will saw below), allowing us to analyze the training dynamic without replacing softmax attention by an artificial surrogate mechanism.

Attention output. Given an input $x = (x_0, x_1, x_2)$, we form three queries $(z_0(x), z_1(x), z_2(x))$. For each head $h \in \{1, 2, 3\}$ and query $z_j(x)$, we use the standard scaled dot-product attention over the corresponding memory pair $(m_{j,+}, m_{j,-})$. Specificially, for $\sigma \in \{-1, 1\}$, the attention weight $A_{h,j}$ can be calculated as

$$A_{h,j,\sigma}(x) = \frac{\exp\left(\langle Q_h z_j(x), K_h m_{j,\sigma}\rangle / \sqrt{3}\right)}{\Sigma_{\tau\in\{\pm1\}} \exp\left(\langle Q_h z_j(x), K_h m_{j,\tau}\rangle / \sqrt{3}\right)}.$$

Roughly speaking $A_{h,j,+}(x)$ and $A_{h,j,-}(x)$ are simply the two softmax weights assigned by head h to the positive and negative memory token associated with feature j. The corresponding value vectors are $V_h m_{j,+}$ and $V_h m_{j,-}$, and the output $g_{h,j}(x)$ of head h for the query j can be written as

$$g_{h,j}(x) = O_h \left(\sum_{\sigma\in\{-1,1\}} A_{h,j,\sigma}(x) V_h m_{j,\sigma} \right) \in \mathbb{R}^3.$$

We then sum over all feature queries and all three attention heads: $g(x) = \sum_{h=1}^{3} \sum_{j=0}^{2} g_{h,j}(x) \in \mathbb{R}^3$. Because our prediction target Y is scalar, we finally apply the fixed readout $r(z) = \mathbf{1}^\top z$, and the scalar prediction of the model is $f_\theta(x) = \mathbf{1}^\top g(x)$, where θ denotes the tuple containing all the trainable parameters Q_h, K_h, V_h, O_h, for $h \in \{1, 2, 3\}$.

Diagonal parameterization. Following prior works of Transformer training dynamics (e.g., (Boix-Adsera et al., 2023)), for simplicity and tractability of the analysis, we restrict the query, key, value, and output projection matrices of each head to be diagonal, that is

$$Q_h = \mathrm{diag}(q_{h0}, q_{h1}, q_{h2}), K_h = \mathrm{diag}(k_{h0}, k_{h,1}, k_{h2}),$$
$$V_h = \mathrm{diag}(v_{h0}, v_{h1}, v_{h2}), O_h = \mathrm{diag}(o_{h0}, o_{h1}, o_{h2}).$$

This restriction removes cross-coordinate interactions and allows us to focus on analyzing the competition among heads for each features. Importantly, we note that this assumption does NOT make the optimization dynamic linear, as the query, key, value, and output parameters are still trained separately and interact multiplicatively through the softmax attention mechanism.

Training objective. Since our goal is to understand whether joint training can create the capability enhancement localization phenomenon, we train the same attention model jointly on task A and B. For task $T \in \{A, B\}$, let $R_T(\theta) = \mathbb{E}_T[(f_\theta(X) - Y)^2]$ denote the population squared risk of the attention model on task T, where f_θ is scalar predictor defined by the attention block above. Since the two considered tasks are equally weighted, we train the predictor f_θ using the half-squared population loss, that is $\mathcal{L}(\theta) = \frac{1}{4}(R_A(\theta) + R_B(\theta)$.

Weights initialization and training algorithm. We train all query, key, value, and output parameters using fixed step-size *full-batch* gradient descent. Concretely, for each head $h \in \{1, 2, 3\}$ and feature j, we collect the four trainable scalar parameters into $\theta_{hj} = (q_{hj}, k_{hj}, v_{hj}, o_{hj})$. We initialize the weights θ_{hj} as θ^0_{hj}, where

$$\theta^0_{hj} = \alpha G_{hj}, \quad G_{hj} \overset{i.i.d}{\sim} N(0, I_4),$$

where $\alpha > 0$ is a small positive constant controlling the initialization scale. In other words, the weights θ_{hj} at initialization has covariance $\alpha^2 I_4$. Starting from this initialization, all parameters are updated using full-batch GD

$$\theta^{n+1} = \theta^n - \eta \nabla \mathcal{L}(\theta^n),$$

where $\eta > 0$ is fixed learning rate. Importantly, no feature is assigned to any head beforehand, and we do NOT perform sign conditioning, parameter tying, re balancing, or any other intervention mechanism during training. This means that *if a particular feature eventually becomes concentrated in one attention head, that localization must arise naturally from the training dynamics and the random initialization rather than from predetermined head assignment.*

4.2 Joint Optimality Does Not Determine Head Allocation

We first express the attention model in terms of its effective feature coefficients. This representation identifies the predictor that minimizes the joint population loss and separates this aggregate optimum from the allocation of features across individual heads.

Proposition 4.1 (Exact softmax reduction and coefficient compromise)**.** *Consider the tree-head attention model described as in Section 4.1, with the fixed coordinate encoding, paired memory bank, and diagonal query, key, value and output projection matrices. For each head $h \in \{1, 2, 3\}$ and feature $j \in \{0, 1, 2\}$, define the effective head-feature coefficient $c_{hj} = o_{hj} v_{hj} \tanh(q_{hj} k_{hj})$, and let $w_j = \sum_{h=1}^{3} c_{hj}$ denote the total coefficient assigned to feature j after summing over three heads. Then we have*

1. ***Head-feature response:*** *For each input $x = (x_1, x_2, x_3)$, the contribution of head h associated with feature j is exactly $x_j c_{hj} e_j$.*

2. ***Predictor representation****: The model prediction can be written as $f_\theta(x) = w^\top x$, where $w = (w_0, w_1, w_2)$ and w_i is defined as above. In other words, the resulting predictor is linear in the input features. However, the effective coefficients w_j are highly non-linear functions of the trainable weights.*

3. ***Population risks:*** *Under the two data distributions defined as in Section 4.1, the population risks reduce to*

$$R_A(w) = w_0^2 + (w_1 - b)^2, \quad R_B(w) = (w_0 - a)^2 + (w_2 - c)^2.$$

4. ***Joint training objective and optimum:*** *As a result, the population objective $\mathcal{L}(\theta) = \frac{1}{4}\left(R_A(w(\theta)) + R_B(w(\theta))\right)$ can be rewritten as*

$$\mathcal{L}(\theta) = \frac{1}{2}\left(w_0 - \frac{a}{2}\right)^2 + \frac{1}{4}(w_1 - b)^2 + \frac{1}{4}(w_2 - c)^2 + \frac{a^2}{8}.$$

   This implies the optimal aggregate coefficient vector $w^\star = \beta = \left(\frac{a}{2}, b, c\right)$.

Proof sketch. The proof of Proposition 4.1 is simple which only involves in basic transformation and the original problem setting (Section 4.1). The detailed proof can be found in Appendix B. □

Task conflict produces a compromise. Within this model, the coefficient on (U) is shared across tasks and cannot depend on which task generated the input. Task (A) requires coefficient zero, whereas task (B) requires coefficient (a). Joint optimization therefore selects $w_0^\star = a/2$, while fitting the task-specific coefficients (b) and (c) exactly. Both tasks have risk $a^2/4$ at this joint optimum.

Head allocation remains undetermined. Although the optimal aggregate vector is unique, its decomposition across heads is not: the objective constrains only the sums $w_j = \sum_h c_{hj}$. This distinction matters because ablating head h changes the predictor's coefficient vector from w to

$w - c_h$, where $c_h = (c_{h0}, c_{h1}, c_{h2})$. Removing a head that carries the conflicting coordinate may therefore also remove useful task-specific signal. Establishing beneficial head ablation requires understanding which decomposition training selects. We next show that gradient descent from small random initialization concentrates each feature in a single head.

4.3 Gradient descent localizes features

We now show that gradient descent selects a particular structure among the many head decompositions compatible with the optimal aggregate predictor. Under small Gaussian initialization, each feature becomes concentrated in one head, while its contributions from the remaining heads stay small throughout training.

Theorem 4.2 (Training-induced feature localization)**.** *For $h \in \{1,2,3\}$, $j \in \{0,1,2\}$, let*

$$\theta_{hj} = (q_{hj}, k_{hj}, v_{hj}, o_{hj}), \quad c_{hj} = o_{hj} v_{hj} \tanh(q_{hj} k_{hj}), \quad w_j = \sum_{h=1}^{3} c_{hj}.$$

Fix $a, b, c > 0$, set $\beta = \left(\frac{a}{2}, b, c\right)$ and $p = \left(1, \frac{1}{2}, \frac{1}{2}\right)$, and train $\mathcal{L}(\theta) = \frac{1}{2}\sum_{j=0}^{2} p_j (w_j - \beta_j)^2$ by full-batch GD from the initialization $\theta_{hj}^0 = \alpha G_{hj}$, where $G_{hj} \overset{i.i.d.}{\sim} N(0, I_4)$. Then there exists $\eta_\star > 0$, depending on only a, b, c, such that the following holds for every fixed step size $0 < \eta < \eta_\star$. For almost every draw of G, all sufficiently small $\alpha > 0$, we have:

1. ***Convergence:*** *the parameter trajectory is bounded and converges to a finite limit with $w^\infty = \beta$. Each residual $r_j^n = \beta_j - w_j^n$ is positive and non-increasing.*
2. ***Localization:*** *there are labels $h_j = h_j(G)$, independent and uniform on $\{1,2,3\}$, and a finite constant K_G, independent of α, such that*

$$\sup_{n \geq 0} \|\theta_{hj}^n\|_\infty \leq K_G \alpha, \quad \sup_{n \geq 0} |c_{hj}^n| \leq K_G \alpha^4 \text{ for } h \neq h_j.$$

   Consequently, $c_{h_j j}^\infty = \beta_j + \mathcal{O}_G(\alpha^4)$.
3. ***Finite checkpoint:*** *define $N_\alpha = \lceil \frac{\log(1/\alpha)}{\eta \alpha^2} \rceil$, for every fixed $M > 0$ and sufficiently small α, we have $\sup_{n \geq N_\alpha} \max_j r_j^n \leq \alpha^M$.*

The proof details of Theorem 4.2 can be found in Appendix B. However, we can temporarily forget about the proof of this result and instead focus on its interpretation. Theorem 4.2 establishes localization through training without prescribing a feature-to-head assignment. Every head initially has access to every feature, but random initialization and gradient descent select a winning head for each feature. This is a feature-wise statement: different features may share the same winning head. Thus, localization alone does not yet guarantee beneficial head ablation. The next subsection identifies an allocation event under which removing the head carrying the conflicting feature preserves the task-specific features and improves task A.

4.4 Localization Enables Beneficial Head Ablation

We now connect feature localization to task-specific improvement. If the conflicting feature is concentrated in a head that carries neither task-specific feature, ablating that head removes the unwanted contribution to task A while approximately preserving its useful signal. The independent winning-head assignments in Theorem 4.3 give this event a positive probability. The detailed proofs of Theorem 4.3 can be found on Appendix B.

Theorem 4.3 (Training-induced beneficial head ablation)**.** *Under the assumptions of Theorem 4.2, let $t = a/2$, and define the event*

$$E = \{h_0 \notin \{h_1, h_2\}\},$$

where h_j is the head that wins the feature $j \in \{0,1,2\}$. Then $\Pr(E) = \frac{4}{9}$. Under such event, set $h_\star = h_0$, and delete this head without changing any other parameters

$$f_{-h_\star}^n = f^n - f_{h_\star}^n.$$

For almost every initialization draw G in E, all sufficiently small $\alpha > 0$, and every $n \geq N_\alpha$,

$$G_A^n \triangleq R_A(f^n) - R_A(f^n_{-h_\star}) = t^2 + \mathcal{O}_G(\alpha^4) > 0,$$
$$D_B^n \triangleq R_B(f^n_{-h_\star}) - R_B(f^n) = 3t^2 + \mathcal{O}_G(\alpha^4) > 0,$$

where the error bound are uniform over $n \geq N_\alpha$. Consequently

$$\liminf_{\alpha \to 0^+} \Pr_G \left(\exists h : R_A(f^{N_\alpha}) - R_A(f^{N_\alpha}_{-h}) \frac{t^2}{2}, R_B(f^{N_\alpha}_{-h}) - R_B(f^{N_\alpha}) \geq \frac{3t^2}{2} \right) \geq \frac{4}{9}.$$

Interpretation of Theorem 4.3. In the small-initialization limit, both intact task risks approach t^2. Removing the selected head reduces task A's risk to zero but increases task B's risk to $4t^2$. Consequently, the joint objective increases by $t^2/2$. The intervention therefore improves one task by moving away from the jointly optimal compromise. By parameter convergence, the same ablation conclusions also hold at the limiting trained model. This therefore theoretically describes the CEL phenomenon.

5 Discussion and Limitation

Empirical scope. Our experiments provide evidence of beneficial sparse head ablations across three model families and four benchmarks. These findings concern the evaluated models, datasets, and inference protocols; they do not establish that beneficial ablations exist for every model–task pair. Although selection and evaluation use disjoint examples, the stability of the selected subsets and their gains across different discovery samples remains insufficiently characterized.

Search and head interactions. Our two-stage procedure identifies beneficial interventions but does not guarantee an optimal subset. Individual-head screening can exclude combinations whose benefits emerge only through joint ablation. Consequently, the gains depend on both the discovery data and the candidate pool, and an unsuccessful search would not establish the absence of beneficial sparse interventions.

Theoretical scope. We note that our analysis establishes a possible training-induced mechanism for CEL under a fixed feature encoding, diagonal projections, small Gaussian initialization, and full-batch gradient descent on a noiseless population objective. The architecture supplies the feature directions and restricts cross-coordinate interactions, while training determines their allocation across heads. The resulting task conflict arises within the shared predictor class. Therefore, the theory establishes one possible mechanism through which beneficial head ablation can arise despite joint optimality. It does not establish that this mechanism explains the empirical gains in the evaluated language models, nor that it accounts for all instances of CEL.

6 Conclusion and Future Works

In this work, we study Capability Enhancement Localization (CEL): whether removing a small set of attention heads can improve task-specific performance without further fine-tuning. Across three model families and four benchmarks, a simple two-stage search identifies sparse head subsets whose ablation improves held-out accuracy. Motivated by these observations, we establish a potential training-induced mechanism for CEL in a simplified softmax attention model. Gradient descent from small Gaussian initialization concentrates features in individual heads, and, with positive probability, removing the head carrying a conflicting feature improves one task while harming another. This can occur even at a global minimum of the joint training objective, showing that successful joint optimization is compatible with beneficial task-specific head ablation.

We note that this work only provides an initial empirical and theoretical investigation of CEL. On the empirical side, future work could develop more efficient search procedures that account for interactions among heads, including beneficial combinations that individual-head screening may overlook. On the theoretical side, an important direction is to characterize when training produces beneficial head ablations beyond the fixed feature encoding and diagonal parameterization studied here. In particular, allowing learned representations and interactions among features could clarify when conflicting contributions become sufficiently separated from useful ones for sparse ablation to improve task performance.

AI USE STATEMENT

Experiment implementation. For the codes of all experiments described in this work, the authors use OpenAI Codex (with models GPT-5.6 Sol, GPT-6 Astra) for implementations.

Theoretical results. For the theoretical models, claims, and construction provided in this work, the authors use GPT-6.0 Astra for brainstorming ideas, theorem construction, and theorem proving, though the authors already had clue about the constructions at the first place. After that, the authors organize the statements, the proofs, and the elaboration in this paper with the help of LLMs. Finally, when incorporating the statements/proofs/elaborations into the main body and appendix, the authors rewrite the contents based on the authors' understanding. The authors verified the soundness and correctness of the theoretical results presented in this draft.

Paper writing. For paper writing, the authors use GPT-5.6 Sol and Grammarly. For every paragraph in this paper, the authors use the following workflow: (1) the authors draft the paragraph without the use of LLMs, (2) the authors then use GPT-5.6 Sol to improve/correct the wording and grammars, (3) the authors again modify the paragraph (fixed by GPT-5.6 Sol) again to improve the flow and personalize the deliver, and (4) the authors finally use Grammarly to fix the typos and grammars once again. The authors provide the citations and references by hand in the first step using Google Scholar, and no LLM is used for generating citation.

ETHICS STATEMENT

This paper presents work whose goal is to advance the field of AI and Machine Learning. There are many potential societal and ethical consequences of our work, none which we feel must be specifically highlighted here.

REPRODUCIBILITY STATEMENT

All the theoretical claims in this paper are provided with clear explanations of any assumptions and complete proof (which can be in the main body or in Appendix). The code for experiments in this paper is provided in `https://anonymous.4open.science/r/capability-enhancement-localization-DFEB`.

REFERENCES

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Anna Bair, Yixuan Even Xu, Mingjie Sun, and J Zico Kolter. Compressed sensing for capability localization in large language models. *arXiv preprint arXiv:2603.03335*, 2026.

Elie Bakouch, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, Lewis Tunstall, Carlos Miguel Patiño, Edward Beeching, Aymeric Roucher, Aksel Joonas Reedi, Quentin Gallouédec, et al. Smollm3: smol, multilingual, long-context reasoner. *Hugging Face Blog*, 2025.

Raphaël Berthier. Incremental learning in diagonal linear networks. *Journal of Machine Learning Research*, 24(171):1–26, 2023.

Enric Boix-Adsera, Etai Littwin, Emmanuel Abbe, Samy Bengio, and Joshua Susskind. Transformers learn through gradual rank increase. *Advances in Neural Information Processing Systems*, 36: 24519–24551, 2023.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

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, 2019a.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. What does bert look at? an analysis of bert's attention. In *Proceedings of the 2019 ACL workshop BlackboxNLP: analyzing and interpreting neural networks for NLP*, pp. 276–286, 2019b.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352, 2023.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. 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. 4171–4186, 2019.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Peter Hase, Mohit Bansal, Been Kim, and Asma Ghandeharioun. Does localization inform editing? surprising differences in causality-based localization vs. knowledge editing in language models. *Advances in Neural Information Processing Systems*, 36:17643–17668, 2023.

William Held and Diyi Yang. Shapley head pruning: Identifying and removing interference in multilingual transformers. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 2416–2427, 2023.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Xiusheng Huang, Jiaxiang Liu, Yequan Wang, Jun Zhao, and Kang Liu. Capability localization: Capabilities can be localized rather than individual knowledge. In *International Conference on Learning Representations*, volume 2025, pp. 40446–40462, 2025.

Zhuoran Jin, Pengfei Cao, Hongbang Yuan, Yubo Chen, Jiexin Xu, Huaijun Li, Xiaojian Jiang, Kang Liu, and Jun Zhao. Cutting off the head ends the conflict: A mechanism for interpreting and mitigating knowledge conflicts in language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 1193–1215, 2024.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? *Advances in neural information processing systems*, 32, 2019.

Hieu Trung Nguyen, Bao Nguyen, and Viet Anh Nguyen. Structured pruning for diverse best-of-n reasoning optimization. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 23911–23922, 2025.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*, 2022.

Abhishek Panigrahi, Nikunj Saunshi, Haoyu Zhao, and Sanjeev Arora. Task-specific skill localization in fine-tuned language models. In *International Conference on Machine Learning*, pp. 27011–27033. PMLR, 2023.

Scott Pesme and Nicolas Flammarion. Saddle-to-saddle dynamics in diagonal linear networks. *Advances in Neural Information Processing Systems*, 36:7475–7505, 2023.

A Yang Qwen, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengpeng Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, pp. 2412–15115, 2025.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*, 2021.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, 1996.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 5797–5808, 2019.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. *arXiv preprint arXiv:2211.00593*, 2022.

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. *arXiv preprint arXiv:2109.01652*, 2021.

Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. Retrieval head mechanistically explains long-context factuality. In *International Conference on Learning Representations*, volume 2025, pp. 62143–62156, 2025.

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.

Wei Zhang, Chaoqun Wan, Yonggang Zhang, Yiu-ming Cheung, Xinmei Tian, Xu Shen, and Jieping Ye. Interpreting and improving large language models in arithmetic calculation. *arXiv preprint arXiv:2409.01659*, 2024.