WHEN CAN PREFIXES COMPILE LORA? EXACT RESOURCE-CAPPED TESTS FOR FROZEN ATTENTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Can a fixed continuous prefix replace a given low-rank adapter while the attention head stays frozen? In this research, we show that the answer depends on the adapter's target through three conditions. First, observability: at one causal readout, every independent key–value prefix sees the content only through the query, attention partition, and value numerator, so a target that differs on two inputs with equal summaries incurs an error floor at every prefix length; norm caps extend this floor to nearly equal summaries. Second, realizability: at a common query, any prefix reduces exactly to two aggregate variables, and the norm-capped optimum is an attained second-order-cone program, also after a fixed output projection; it places two equal-norm rank-one value updates on opposite sides of compilability. Third, implementation: under affine query exposure, $2r$ signed slots approximate a rank-r value update, but their values grow as $O(\epsilon^{-3/2})$, and the construction passes all 400 tolerance checks in float64 yet only 38 in bfloat16. A first-layer GPT-2 readout with fixed token and position meets the common-query condition without clamping activations; at three such heads, the capped optimum leaves 18.4% to 74.2% of the projected adapter effect uncompiled, with a head-dependent value–query ordering. All claims concern local approximation at one head, not whole-network equivalence.

1 INTRODUCTION

A low-rank adapter changes parameters; a prefix changes the context visible to a frozen attention mechanism (Hu et al., 2022; Li & Liang, 2021). Similar behavior does not mean that one can replace the other, so the practical question is target-specific: given a particular adapter and a fixed model, can one shared prefix reproduce its outputs on a declared input domain? A longer prefix helps only if the missing behavior lies within what the interface can represent.

Two established results leave this question open. On one hand, a prefix preserves relative attention among the existing content tokens and only mixes the frozen output with a query-dependent prefix contribution (Petrov et al., 2024b). On the other hand, prompting can be universal for suitably constructed or pretrained transformers (Wang et al., 2023; Petrov et al., 2024a), including random transformers under appropriate rank conditions (Hsu & Lai, 2026). Yet attention invariance does not decide output emulation, because prefix values can compensate for unchanged attention weights, and universality of a model class does not make a prescribed update emulable at a fixed head.

We therefore study one causal softmax head with its input states, readout, and content positions fixed, and give the prefix its strongest local form: independently chosen keys and values, with one prefix shared across all inputs. Under this interface, compilability depends on the target rather than on whether the adapter is value-side or query-side. Figure 1 organizes our results as three tests that a target must pass. Our contributions are:

- **Observability.** The prefixed response depends on content only through $\Sigma = (q, Z, N)$, a complete intervention statistic. Target variation within an equal-summary pair gives an error floor at every prefix length, and key and value caps extend it to near-equal summaries (Section 4).

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.

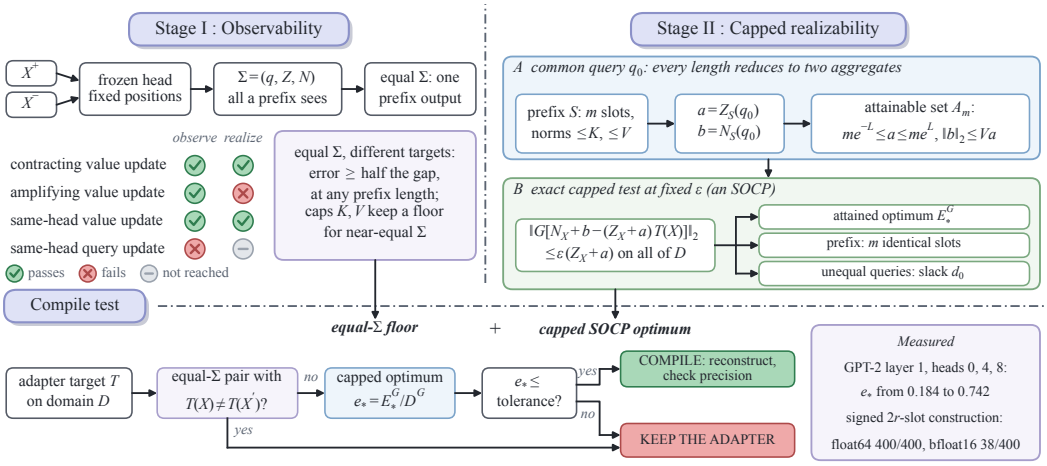


Figure 1: Overview. **Stage I**: a prefix sees the content only through $\Sigma = (q, Z, N)$, so a target that differs on an equal-Σ pair keeps an error floor at every prefix length (Section 4). **Stage II**: at a common query, any prefix reduces to two aggregates, and the capped optimum is an exact SOCP (Section 5). The **compile test** combines both; a compiled prefix must still be checked at the target precision (Section 6). Measured values are from Section 7.

- **Exact capped realizability.** At a common query, every prefix length reduces to two aggregate variables, and norm-capped feasibility is an SOCP with an exact reconstruction, also after a fixed output projection; a proved slack covers unequal queries. Two equal-norm rank-one value updates fall on opposite sides of this test, and a fixed-token first-layer GPT-2 readout meets its common-query condition; there, the test leaves 18.4% to 74.2% of trained adapter effects uncompiled (Sections 5 and 7.4).
- **Construction and precision.** Affine query exposure lets $2r$ signed slots approximate a rank-r value update, with values growing as $O(\epsilon^{-3/2})$. Controlled tests confirm it on all 400 float64 cases but only 38 in bfloat16, and show that extra capped slots can raise the optimal error (Sections 6 and 7).

2 Related Work

Analyses of in-context learning recover learning algorithms or implicit weight updates in linear and simplified settings (Dai et al., 2023; Akyürek et al., 2023; von Oswald et al., 2023), and prefix-tuning and soft-prompt tuning learn continuous context directly (Li & Liang, 2021; Lester et al., 2021). These results motivate converting between context and parameters, but neither direction is automatically invertible for one fixed head and a prescribed adapter. Closest to our question, ReasonCACHE (Gupta et al., 2026) compares the output subspaces that prefixes and LoRA reach at a fixed context. Our requirement is stricter: a single prefix must reproduce one target across many inputs, which a per-input comparison does not enforce.

We build on the prefix decomposition of Petrov et al. (2024b) and do not claim its relative-attention invariant as new, nor the competition for attention mass it implies (Wang et al., 2026). From that interface we extract a complete intervention statistic, a robust target-error bound, and a finite-dimensional realizability class at a common query. The cone formulation uses standard quasiconvex feasibility machinery (Boyd & Vandenberghe, 2004); the attention-specific step is the exact description of attainable prefix mass and numerator. Universality, capacity, and memorization results for prompting (Wang et al., 2023; Petrov et al., 2024a; Hu et al., 2025; Meyer et al., 2025; Hsu & Lai, 2026) rely on different model and exposure conditions, whereas our positive construction targets an exposed linear value update with a rank-dependent slot count. Unlike general attention-sensitivity bounds (Kim et al., 2021; Castin et al., 2024), our bound fixes the content summary and caps prefix keys and values. Appendix B discusses context-to-weight conversion, memorization limits, and parameter-efficient adaptation in more detail.

3 A FIXED ATTENTION INTERFACE

Consider a causal head with $W^Q, W^K \in \mathbb{R}^{d_k \times d}$, $W^V \in \mathbb{R}^{d_v \times d}$, and output projection $W^O \in \mathbb{R}^{d_o \times d_v}$. For $X = (x_1, \ldots, x_j)$ at readout j, define

$$q = W^Q x_j, \qquad k_i = W^K x_i, \qquad v_i = W^V x_i,$$
$$s_i = q^\top k_i / \sqrt{d_k}, \qquad Z_X = \sum_{i \le j} e^{s_i}, \qquad N_X = \sum_{i \le j} e^{s_i} v_i, \qquad h_X = N_X / Z_X. \tag{1}$$

A finite independent key–value prefix is $S = \{(\kappa_t, \nu_t)\}_{t=1}^m$, with $m \ge 1$, whose keys and values are chosen directly rather than derived from a common embedding. Let

$$Z_S(q) = \sum_t e^{q^\top \kappa_t / \sqrt{d_k}}, \qquad N_S(q) = \sum_t e^{q^\top \kappa_t / \sqrt{d_k}} \nu_t. \tag{2}$$

The prefixed head is

$$h_{S,X} = \frac{N_X + N_S(q)}{Z_X + Z_S(q)}. \tag{3}$$

Throughout, content states and their positions stay fixed. This interface contains every key–value pair that coupled soft tokens can realize at the head, but it excludes prefix-induced changes to earlier layers and positional shifts. Our lower bounds therefore apply to coupled head-level soft tokens under the same fixed-state assumptions, not automatically to input prompts in a deep transformer. Affine projection biases are absorbed by augmenting the input with a constant coordinate, while prefix keys and values remain the actual post-projection vectors. For a head inside a multihead block, W^O denotes its output-projection block, and the other heads' contributions stay unchanged.

An adapter defines a target $T_\Delta(X) = h_X^\Delta$, the head output after the update. Value LoRA has $\Delta W^V = BA$, with $B \in \mathbb{R}^{d_v \times r}$ and $A \in \mathbb{R}^{r \times d}$, and query LoRA changes W^Q instead. Because one fixed prefix must serve every X in a nonempty domain $\mathcal{D}$, we measure its uniform error

$$\mathcal{E}_\mathcal{D}(S, T) = \sup_{X \in \mathcal{D}} \left\| h_{S,X} - T(X) \right\|_2. \tag{4}$$

Let $\mathcal{P}_{\mathrm{KV}}^{<\infty}$ denote all such finite nonempty prefixes. A target compiles when $\inf_{S \in \mathcal{P}_{\mathrm{KV}}^{<\infty}} \mathcal{E}_\mathcal{D}(S, T) = 0$, and compiles exactly when the infimum is attained. An input-dependent prefix $S(X)$ falls outside this definition.

4 OBSERVABILITY AND ROBUST ERROR FLOORS

We first ask what a prefix can see of the content.

Lemma 1 (Prefix decomposition; Petrov et al., 2024b)**.** *For $\rho = Z_S(q)/(Z_X + Z_S(q))$ and $g_S(q) = N_S(q)/Z_S(q)$,*

$$h_{S,X} = (1 - \rho) h_X + \rho g_S(q). \tag{5}$$

Consequently, the content is observed only through

$$\Sigma(X) = (q, Z_X, N_X). \tag{6}$$

Proof sketch. The mixture follows by splitting the numerator of (3). We call two inputs with the same Σ an equal-summary pair; they lie in the same Σ-fiber, and every prefix returns one common output on them. Conversely, Σ is exactly what a prefix can probe:

Proposition 2 (Complete intervention statistic)**.** *Two inputs have the same Σ if and only if their outputs agree under every one-slot independent-KV prefix.*

Proof sketch. The forward direction is immediate. For the converse, a zero key with value ν gives the output $(N + \nu)/(Z + 1)$; equality for every ν forces $Z = Z'$ and then $N = N'$. An arbitrary key then forces $e^{q^\top \kappa / \sqrt{d_k}} = e^{q'^\top \kappa / \sqrt{d_k}}$ for every κ, hence $q = q'$. Completeness concerns interventions; it does not make every function of Σ expressible. It does imply that any target separating an equal-summary pair must incur error:

Proposition 3 (Fiber-diameter obstruction). *For any target $T : \mathcal{D} \to \mathbb{R}^{d_v}$,*

$$\inf_{S \in \mathcal{P}_{\mathrm{KV}}^{<\infty}} \mathcal{E}_{\mathcal{D}}(S, T) \geq \frac{1}{2} \sup_{\substack{X, X' \in \mathcal{D} \\ \Sigma(X) = \Sigma(X')}} \|T(X) - T(X')\|_2 . \tag{7}$$

Proof sketch. Apply the triangle inequality around the common prefixed output; the argument holds at any prefix length. On a continuous domain, however, exact collisions can be hard to find or to verify numerically. The next result shows how close two summaries must be once prefix resources are bounded.

Theorem 4 (Length-independent near-fiber obstruction). *Restrict every prefix slot to $\|\kappa_t\|_2 \leq K$ and $\|\nu_t\|_2 \leq V$. For a pair with $\|h_X\|_2, \|h_{X'}\|_2 \leq H$, define*

$$\begin{aligned} \omega(X, X') = \; & \|h_X - h_{X'}\|_2 + \frac{H + V}{4} \left| \log Z_X - \log Z_{X'} \right| \\ & + \left(V + \frac{H + V}{4} \right) \frac{K}{\sqrt{d_k}} \|q - q'\|_2 . \end{aligned} \tag{8}$$

Every prefix of any finite length satisfies $\|h_{S,X} - h_{S,X'}\|_2 \leq \omega(X, X')$. Therefore its worst-case target error on the pair is at least

$$\frac{1}{2} \left[\|T(X) - T(X')\|_2 - \omega(X, X') \right]_+ . \tag{9}$$

Proof sketch. Length independence rests on three facts: $\log Z_S$ is $K/\sqrt{d_k}$-Lipschitz in q; g_S has Jacobian $\mathrm{Cov}(\nu, \kappa)/\sqrt{d_k}$ with operator norm at most $VK/\sqrt{d_k}$; and the mixing weight is a sigmoid of $\log Z_S - \log Z_X$, with derivative at most $1/4$. Combining them with (5) gives (8), and the triangle inequality gives (9) (Appendix D). The caps are essential, since the theorem says nothing about prefixes whose magnitudes grow without bound.

5 Observable targets need not be realizable

Equal summaries are not the only obstruction. Even when the summary fully determines the target, softmax normalization can prevent any single prefix from realizing it. At a common query, the optimum can be characterized exactly rather than only bounded.

Theorem 5 (Exact common-query reduction). *On a finite domain with a common nonzero query q_0, the optimal prefix error is*

$$\inf_{S \in \mathcal{P}_{\mathrm{KV}}^{<\infty}} \mathcal{E}_{\mathcal{D}}(S, T) = \inf_{a > 0,\, b \in \mathbb{R}^{d_v}} \max_{X \in \mathcal{D}} \left\| \frac{N_X + b}{Z_X + a} - T(X) \right\|_2 . \tag{10}$$

Every (a, b) on the right is realized by one slot. If, additionally, $Z_X = Z_0$ is constant, the closure of prefix outputs on the domain is exactly

$$h_{S,X} = t h_X + c, \qquad 0 \leq t \leq 1, \quad c \in \mathbb{R}^{d_v}. \tag{11}$$

For a two-input domain, the resulting minimax error is

$$\frac{1}{2} \min_{0 \leq t \leq 1} \left\| T(X^+) - T(X^-) - t(h_{X^+} - h_{X^-}) \right\|_2 . \tag{12}$$

Proof sketch. Any prefix supplies $a = Z_S(q_0) > 0$ and $b = N_S(q_0)$; conversely, the single slot $\kappa = \sqrt{d_k} \log(a) q_0 / \|q_0\|^2$, $\nu = b/a$ realizes any such pair. With a common partition, $t = Z_0/(Z_0 + a)$ and $c = b/(Z_0 + a)$ give every $0 < t < 1$ with any translation c, and limits add the endpoints. For two inputs, the best translation aligns the midpoints of targets and predictions, leaving half of the difference mismatch. Some of these infima are not attained by finite parameters.

At fixed tolerance ϵ, the constraints $\|N_X + b - (Z_X + a) T(X)\|_2 \leq \epsilon (Z_X + a)$ are second-order-cone constraints, so varying partitions do not prevent a convex feasibility test. The unconstrained infimum needs care because of strict positivity and unbounded mass, and Appendix F.3 gives a

compactified formulation. A zero query is a separate boundary case, since every prefix key then has unit mass. Explicit resource caps remove these difficulties and yield an attained optimum. $\|N_X + b - (Z_X + a)T(X)\|_2 \le \epsilon(Z_X + a)$ are second-order-cone constraints. Thus varying partitions do not prevent a convex feasibility test. Strict positivity and unbounded mass require care for an unconstrained infimum; Appendix F.3 gives a compactified formulation. The next result supplies an attained optimum under explicit resource caps. A zero query is a separate boundary because every prefix key then has unit mass.

5.1 Exact resource-capped feasibility

Theorem 6 (Capped aggregate characterization)**.** *Fix a finite nonempty domain with common query $q_0 \neq 0$, exactly $m \ge 1$ prefix slots, and caps $\|\kappa_t\|_2 \le K$, $\|\nu_t\|_2 \le V$, where $K, V \ge 0$. Set $L = K\,\|q_0\|_2 / \sqrt{d_k}$. The attainable pairs $(a, b) = (Z_S(q_0), N_S(q_0))$ are exactly*

$$\mathcal{A}_m = \{(a, b) : me^{-L} \le a \le me^{L}, \quad \|b\|_2 \le Va\}. \tag{13}$$

For a fixed linear map G on head outputs, the minimum projected target error is attained and is the smallest $\epsilon \ge 0$ for which

$$(a, b) \in \mathcal{A}_m, \qquad \|G[N_X + b - (Z_X + a)T(X)]\|_2 \le \epsilon(Z_X + a) \quad \textit{for all } X \tag{14}$$

is feasible. At fixed ϵ, this is an SOCP. Every feasible (a, b) has an exact realization using m identical slots:

$$\kappa_t = \frac{\sqrt{d_k}\log(a/m)}{\|q_0\|_2^2} q_0, \qquad \nu_t = b/a. \tag{15}$$

For $q_0 = 0$, replace the mass interval by $a = m$; zero keys and the same values realize every admissible pair.

Proof sketch. Necessity holds because each key contributes mass in $[e^{-L}, e^{L}]$ and the value cap gives $\|b\| \le Va$; the reconstruction (15) gives sufficiency, and compactness of $\mathcal{A}_m$ with positive denominators gives attainment (Appendix F, with solver details). Taking $G = I$ evaluates the head itself, whereas $G = W^O$ tests whether an obstruction survives the actual output projection; G need not be invertible.

With exactly m slots, some prefix mass is unavoidable. With at most m slots, one minimizes over $1 \le k \le m$, adding the unprefixed output when zero slots are allowed, so the optimum cannot worsen as m grows. The next example shows that realizability alone can decide compilability.

Corollary 7 (Equal-norm value updates on opposite sides)**.** *Take $x^{\pm} = (1, \pm 1)$, $W^Q = [1\ 0]$, $W^K = 0$, and $W^V = [0\ 1]$ at a one-token readout. The rank-one updates $\Delta W_-^V = [0\ -1/2]$ and $\Delta W_+^V = [0\ 1/2]$ have the same norm and unprefixed error $1/2$. Both targets are functions of Σ. The first compiles exactly, whereas the second has minimax prefix error $1/2$ at every finite prefix length.*

Proof sketch. Here $q = Z = 1$, $h^{\pm} = \pm 1$, and the targets are $\pm 1/2$ or $\pm 3/2$. A zero-key, zero-value slot realizes the contraction $t = 1/2$, and (12) gives the amplification floor. The fiber-diameter bound is zero for both updates because the frozen numerators differ. Projection identity, update norm, frozen error, predictability from Σ, and scalar output subspace all coincide; only realizability differs.

Key caps also change how prefix length behaves. In the same head, $|\kappa_t| \le K$ gives $a \in [me^{-K}, me^{K}]$, so the attainable contraction interval is

$$t \in \left[\frac{1}{1 + me^{K}}, \frac{1}{1 + me^{-K}}\right]. \tag{16}$$

For a symmetric target $T(x^{\pm}) = \pm a_0$, the exact best error is the distance from a_0 to this interval, attained with zero values. Under a fixed per-slot key cap, more slots can therefore make the optimum worse, because their unavoidable attention mass attenuates the frozen output.

5.2 Unequal queries and a pretrained first-layer interface

The common-query condition is restrictive, but a pretrained model can meet it without any change to its weights. Suppose the first attention block receives states $x_i = \phi(E[t_i] + P[i])$, where ϕ is

deterministic and tokenwise, as in the evaluation-mode embedding and pre-attention normalization path of GPT-2 (Hugging Face, 2024). Fixing the readout token and its position then makes x_j, and hence the frozen query, identical across preceding contents, while the partition and value numerator still vary. Theorem 6 therefore applies without replacing learned weights or clamping a computed query, provided the query is nonzero or the zero-query branch is used.

Beyond the first layer, queries generally differ across inputs, and a reference query gives a controlled reduction. Let $H = \max_X \|h_X\|_2$, choose any q_0, and keep each (Z_X, N_X) unchanged when forming the common-query reference. Define

$$d_0 = \left(V + \frac{H+V}{4}\right) \frac{K}{\sqrt{d_k}} \max_X \|q_X - q_0\|_2 . \tag{17}$$

Proposition 8 (Robust common-query reference)**.** *Let E_*^G be the optimal projected error on the original finite domain under the fixed (m, K, V) caps, and let E_0^G be the optimum of* (14) *for the reference summaries. Then*

$$[E_0^G - \|G\|_{\mathrm{op}}\, d_0]_+ \le E_*^G \le E_0^G + \|G\|_{\mathrm{op}}\, d_0. \tag{18}$$

The reconstructed reference-optimal prefix has error at most the displayed upper bound on the original domain.

Proof sketch. The proof changes only the query argument of the prefix term in (5) and reuses the sensitivity constants of Theorem 4; the reference summaries need not come from a clamped model. When queries are widely dispersed, the lower bound can be zero, an inconclusive outcome rather than evidence of compilation.

5.3 A DIRECT PRETRAINED-HEAD EVALUATION

Together, these results define a direct evaluation of a trained adapter. The test needs only a finite input domain; the adapter need not improve any downstream task. We cache the frozen inputs, queries, keys, and values, compute the adapted target on the same inputs, and take G to be the selected head's output-projection block. The unprefixed effect and normalized optimum are

$$D_{\mathcal{D}}^G = \max_{X \in \mathcal{D}} \|G[h_X - T(X)]\|_2, \qquad e_*^G = E_*^G / D_{\mathcal{D}}^G, \tag{19}$$

when $D_{\mathcal{D}}^G > 0$; a zero effect is reported separately rather than divided by a numerical floor. The exact capped program, its reconstructed prefix, and a gradient-trained prefix share the domain, target, caps, and slot count, so their difference measures optimization error, while comparison with $D_{\mathcal{D}}^G$ measures how much of the adapter effect remains uncompilable under those caps.

Solving on the evaluation targets gives the oracle finite-domain optimum, whereas fitting one prefix on separate inputs and freezing it measures transfer. For query adapters at a fixed readout state, only BAx_j is visible, so a larger stored rank does not by itself add input-dependent query directions, and we report the realized query displacement rather than nominal rank. Appendix J specifies the extraction, structural projection slices, caps, and conic residual checks, and Section 7.4 reports the results.

6 A CONSTRUCTIVE BOUNDARY FOR VALUE AND QUERY UPDATES

The tests above are interface conditions, not universal negative statements: a prefix can compile a value update when the query exposes the coordinates the adapter reads. Under affine query exposure, a constant one-token self-score, and bounded adapter coordinates, the signed construction of Theorem 11 (Appendix G) approximates a rank-r value update with $2r$ slots, although its values grow as $O(\epsilon^{-3/2})$; Section 7 measures this precision cost. The following contrast shows, on one head, that success depends on the target rather than on the adapter's side.

6.1 AN EXACT SAME-HEAD CONTRAST

Theorem 9 (Query-side separation and exact value compilation)**.** *There is one fixed head and two three-token inputs with the same readout for which a rank-one value update compiles exactly into*

Table 1: Direct $2r$-slot construction, rank two. Each error is the mean over 20 seeds of the maximum over 10,004 test points. These are not standard errors or uniform-supremum estimates. The last column is the mean maximum prefix-value norm. Full seed-level results and sample SDs accompany the numerical data.

Tolerance	Float64 error	Float32 error	Bfloat16 error	Value norm
10^{-1}	4.27×10^{-2}	4.27×10^{-2}	9.32×10^{-2}	2.38×10^{2}
10^{-2}	4.39×10^{-3}	4.39×10^{-3}	1.93×10^{-1}	7.30×10^{3}
10^{-3}	4.40×10^{-4}	4.37×10^{-4}	1.10×10^{0}	2.30×10^{5}
10^{-4}	4.40×10^{-5}	8.29×10^{-5}	1.32×10^{0}	7.28×10^{6}
10^{-6}	4.40×10^{-7}	1.09×10^{-3}	1.32×10^{0}	7.28×10^{9}

one slot, while a rank-one query update has minimax error

$$\eta_\alpha = \frac{2\sinh\alpha}{2\cosh\alpha+1}, \qquad \alpha > 0, \tag{20}$$

at every finite prefix length.

Proof sketch. The witness uses $d = 3, d_k = 2, d_v = 1, W^Q = e_1e_3^\top, W^K = e_2e_1^\top, W^V = e_2^\top$, and

$$X^+ = (e_1 + e_2, -e_1 - e_2, e_3), \qquad X^- = (-e_1 + e_2, e_1 - e_2, e_3).$$

Both inputs have summary $(e_1, 3, 0)$. The query update $\Delta W^Q = \sqrt{2}\alpha e_2e_3^\top$ yields targets $\pm\eta_\alpha$, so the common prefix output errs by at least η_α, and zero prefix values attain this floor. The value update $\Delta W^V = ce_3^\top$, by contrast, gives both targets $c/3$, which the slot $(0, 4c/3)$ realizes exactly. At $\alpha = 2$ (Figure 3a), the floor is 0.8509, half the target gap of 1.7019.

This contrast does not imply a universal value–query ordering. In a separate rank-one value counterexample with $W^Q = W^K = W^V = 0$, inputs $(\pm e_1, e_2)$, and $\Delta W^V = [1\ 0]$, the summaries are equal but the targets are $\pm1/2$. Query and value adapters can therefore both fail observability, and Corollary 7 shows that a value adapter can also fail after passing it. Corollary 13 carries both statements through W^O.

7 Empirical evaluation

We test four questions derived from the theory. **Q1:** does the signed construction achieve its stated tolerance under verified exposure? **Q2:** does realizability distinguish matched value updates? **Q3:** does a robust floor remain informative once exact summary equality is removed? **Q4:** how much of a trained adapter effect does the capped test leave at pretrained heads? Q1 to Q3 use explicitly defined heads, so observability is controlled rather than inferred from a task label.

7.1 Q1: Direct construction and finite precision

We use $x = (1, z)$, $z \in [-1, 1]^r$, $W^Q = I$, $W^K = 0$, and ranks $r \in \{1, 2, 4, 8\}$, with a random unit base value and B scaled to operator norm one. Across 20 seeds, we evaluate the analytic $2r$-slot prefix, which needs no training, on 10,000 uniform test points plus every cube vertex, reusing the same points across five tolerances and three precisions. The analytic bound is uniform over the cube, whereas the measured maximum covers only the finite test set.

In float64, all $400/400$ rank–seed–tolerance cases pass the sampled tolerance check. At rank two, reducing the tolerance from 10^{-2} to 10^{-6} lowers the float64 mean sampled maximum from 4.39×10^{-3} to 4.40×10^{-7}, while the mean maximum value norm grows from 7.30×10^3 to 7.28×10^9, at the predicted $\epsilon^{-3/2}$ rate. Lower precision breaks this limit: the float32 error at 10^{-6} is 1.09×10^{-3}, and bfloat16 passes only $38/400$ cases, falling from 20/100 at rank one to 0/100 at rank eight (Figure 2). A constant prefix length therefore does not imply stable finite-precision compilation. The failure concerns this implementation of the signed construction, not the existence theorem or every possible prefix.

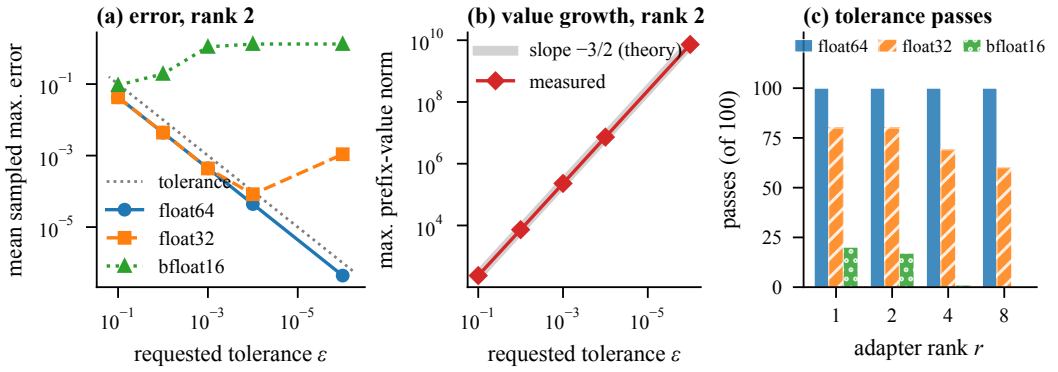


Figure 2: Finite precision and the signed construction. (a) Mean sampled maximum error at rank two over 20 seeds; the dotted line is the requested tolerance. (b) Mean maximum prefix-value norm against the $\epsilon^{-3/2}$ rate of Theorem 11. (c) Tolerance passes per rank among 100 seed–tolerance cases. The prefix keeps $2r$ slots throughout.

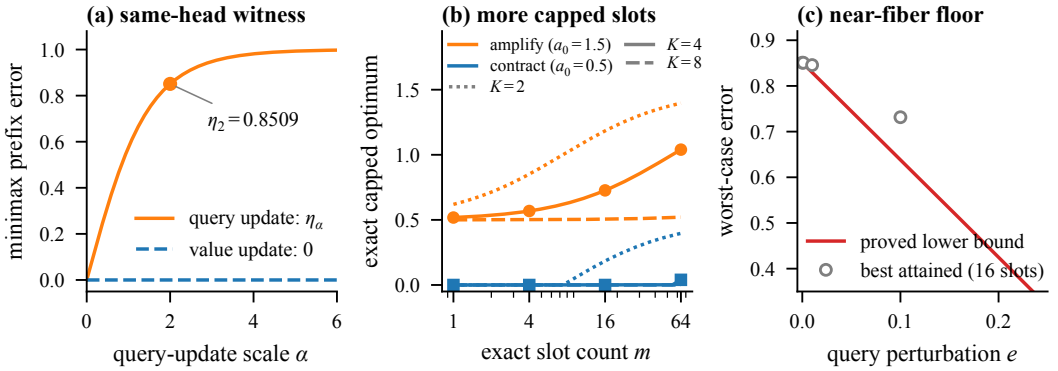


Figure 3: Structural obstructions. (a) Same-head witness of Theorem 9: the rank-one query update has minimax error η_α, while the value update compiles exactly. (b) Exact capped optimum of Corollary 7 against the exact slot count; markers are Table 5 ($K = 4$). (c) Near-fiber lower bound of Theorem 4 with the best attained errors of Table 2.

7.2 Q2: Matched target effects and the cost of extra slots

The contraction and amplification targets of Corollary 7 share the frozen head, rank, update norm, and unprefixed error; both are determined by Σ, and both have a zero fiber-diameter bound. Their bounded-key comparison is the scalar case of Theorem 6. We compute the optimum from (16) and instantiate an attaining prefix with common keys and zero values. Across three key caps and four lengths (24 target–cap–length cases), direct attention evaluation matches the formula within 10^{-12}.

At key cap four (Table 5), one slot realizes the contraction exactly, but 64 slots cannot do better than 0.0396; for the amplifying target, the optimum rises from 0.5180 to 1.0396. Optimization plays no role: a fixed norm cap keeps each added slot's attention mass bounded away from zero. Without the cap, the contraction compiles and the amplification keeps the exact floor $1/2$ at every length. Figure 3b extends this to all three caps: at $K = 2$ even the contraction stops compiling from eight slots on, whereas at $K = 8$ the amplification stays within 0.03 of its uncapped floor up to 64 slots.

7.3 Q3: Near-collisions with optimized bounded prefixes

We next perturb the query witness in a fourth input coordinate, so that $q^{\pm} = (1, \pm e)$ while the frozen partitions and numerators remain equal. The adapted targets become $\eta_{2+e/\sqrt{2}}$ and $-\eta_{2-e/\sqrt{2}}$, and for every $e > 0$ the exact-fiber bound is zero because the queries differ. Theorem 4 still gives a positive lower bound with key cap $K = 2$ and value cap $V = 1$.

Table 2: Near-fiber experiment with 16 slots, key norm at most two, and value norm at most one. The last column is the mean ± sample SD of best-iterate errors over ten optimizer initializations on the same pair, not independent datasets.

Query perturbation e	Proved lower bound	Best attained error	Mean attained error
0	0.8509	0.8509	0.8510 ± 0.0001
0.001	0.8488	0.8507	0.8508 ± 0.0000
0.01	0.8297	0.8457	0.8467 ± 0.0005
0.1	0.6384	0.7315	0.7346 ± 0.0040

To compare this bound with what prefixes attain, we optimize a shared 16-slot prefix on the two inputs with projected Adam (1000 steps, ten initializations), minimizing the worst absolute error over the pair. Attained errors give an upper reference on the optimum, not a proof that the bound is sharp (Table 2). At $e = 0.01$, the bound is 0.8297 and the best attained error 0.8457. At $e = 0.1$, the bound is still 0.6384 despite unequal summaries, and over a continuum of perturbations it decays linearly, staying positive until $e \approx 0.398$ (Figure 3c). The information restriction is therefore robust: additional bounded slots cannot remove it.

7.4 Q4: Pretrained first-layer heads

Finally, we apply the capped test of Section 5.3 to value and query adapters of ranks one and four at heads 0, 4, and 8 of the first GPT-2 attention block, with exactly four slots, the head's output block $G = W^O$, and 128 fitting and 128 evaluation contexts (Appendix J). The conic optimum leaves between 18.4% and 74.2% of the projected adapter effect uncompiled (Appendix Figure 4a), and which side compiles better depends on the head: at rank one, head 0 leaves 0.184 of a value effect and 0.631 of a query effect, whereas head 4 leaves 0.683 and 0.267. Rank four raises the residual fraction in five of the six head–target pairs. The best of ten learned prefixes stays above the conic optimum by only 0.025 to 0.040 of the effect, so the residual reflects the resource limit rather than optimizer error. On held-out contexts (Figure 4b), a conic prefix fitted only on fitting inputs adds 0.065 to 0.067 over the evaluation oracle, and a learned prefix adds a further 0.047 to 0.051.

8 Limitations

All results concern one head under a fixed-state, fixed-position independent key–value interface; changes to other heads, earlier layers, or positions may alter an obstruction, and a head-level floor need not reach the output tokens. The capped test is exact only at a common query, and its reference-query slack can be uninformative elsewhere. The signed construction requires affine query exposure and a constant self-score. The pretrained comparison covers three first-layer GPT-2 heads, two ranks, and fixed caps. We leave prefixes that also change earlier layers to future work.

9 Discussion and Conclusion

We study when one shared prefix can replace a given low-rank adapter at a frozen attention head, and answer with three tests decided per target (Figure 1). Observability reduces to a summary of three quantities beyond which no prefix can distinguish inputs; realizability at a common query becomes an exact, attained second-order-cone program, also after the output projection; and under query exposure, $2r$ signed slots compile a rank-r value update to any tolerance. The tests explain why the adapter's side does not decide the outcome: equal-norm value updates fall on opposite sides, and the value–query ordering changes across pretrained heads, which turns the relative-attention invariant of Petrov et al. (2024b) into a target-level criterion. Because a fixed first-layer readout meets the common-query condition in GPT-2, the exact test applies to pretrained heads without modifying their weights and separates the resource limit from optimizer error. In practice, a compiler can run the test on the target and resource budget of interest, check held-out transfer separately, and compile the adapter into a prefix when both pass. Future work should extend the exact test beyond a common query and ask whether soft tokens or natural demonstrations reach the capped optimum.

REPRODUCIBILITY STATEMENT

The appendices give complete proofs, witness matrices, the capped feasibility and reconstruction procedure, and every controlled experimental setting. The first-layer argument is derived from the pinned architecture implementation. The accompanying numerical scenario file fixes every assumed adapter effect and approximation error and checks their normalizations and ordering. These inputs are not extracted pretrained measurements. All 400-case construction results and the bounded and near-fiber tables retain their stated numerical units and replication structure.

AI USE STATEMENT

Generative AI assisted with literature discovery, mathematical exposition, code preparation, numerical analysis, and language editing. The authors remain responsible for the claims, experimental interpretation, and final submission.

REFERENCES

Kwangjun Ahn, Xiang Cheng, Hadi Daneshmand, and Suvrit Sra. Transformers learn to implement preconditioned gradient descent for in-context learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 45614–45650, 2023.

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? investigations with linear models. In *International Conference on Learning Representations*, 2023.

Yu Bai, Fan Chen, Huan Wang, Caiming Xiong, and Song Mei. Transformers as statisticians: Provable in-context learning with in-context algorithm selection. In *Advances in Neural Information Processing Systems*, volume 36, pp. 57125–57211, 2023.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901, 2020.

Valérie Castin, Pierre Ablin, and Gabriel Peyré. How Smooth Is Attention?, 2024.

Brian K. Chen, Tianyang Hu, Hui Jin, Hwee Kuan Lee, and Kenji Kawaguchi. Exact conversion of in-context learning to model weights in linearized-attention transformers. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 6833–6846, 2024.

Damai Dai, Yutao Sun, Li Dong, Yaru Hao, Shuming Ma, Zhifang Sui, and Furu Wei. Why can GPT learn in-context? language models secretly perform gradient descent as meta-optimizers. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 4005–4019, 2023.

Benoit Dherin, Michael Munn, Hanna Mazzawi, Michael Wunder, and Javier Gonzalvo. Learning without training: The implicit dynamics of in-context learning. *arXiv preprint arXiv:2507.16003*, 2025.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. A survey on in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1107–1128, 2024.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for Transformer Circuits. *Transformer Circuits Thread*, 2021.

Shivam Garg, Dimitris Tsipras, Percy S Liang, and Gregory Valiant. What can transformers learn in-context? a case study of simple function classes. In *Advances in Neural Information Processing Systems*, volume 35, pp. 30583–30598, 2022.

Sharut Gupta, Phillip Isola, Stefanie Jegelka, David Lopez-Paz, Kartik Ahuja, Mark Ibrahim, and Mohammad Pezeshki. ReasonCACHE: Teaching LLMs to reason without weight updates. *arXiv preprint arXiv:2602.02366*, 2026.

Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey. *Transactions on Machine Learning Research*, 2024.

Roee Hendel, Mor Geva, and Amir Globerson. In-context learning creates task vectors. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 9318–9333, 2023.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2790–2799, 2019.

Alexander Hsu and Rongjie Lai. Training-free universal approximation by prompting random transformers. *arXiv preprint arXiv:2608.09558*, 2026.

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.

Jerry Yao-Chieh Hu, Wei-Po Wang, Ammar Gilani, Chenyang Li, Zhao Song, and Han Liu. Fundamental limits of prompt tuning transformers: Universality, capacity and efficiency. In *International Conference on Learning Representations*, pp. 29634–29686, 2025.

Hugging Face. GPT-2 model implementation, Transformers version 4.47.1. Versioned source code, 2024.

Joonseong Kang, Soojeong Lee, Subeen Park, Sumin Park, Taero Kim, Jihee Kim, Ryunyi Lee, and Kyungwoo Song. Adaptive task vectors for large language models. *arXiv preprint arXiv:2506.03426*, 2025.

Hyunjik Kim, George Papamakarios, and Andriy Mnih. The Lipschitz Constant of Self-Attention. In *International Conference on Machine Learning*, volume 139, pp. 5562–5571, 2021.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3045–3059, 2021.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4582–4597, 2021.

Yingcong Li, Muhammed Emrullah Ildiz, Dimitris Papailiopoulos, and Samet Oymak. Transformers as algorithms: Generalization and stability in in-context learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19565–19594, 2023.

Zeju Li, Yizhou Zhou, and Qiang Xu. Latent context compilation: Distilling long context into compact portable memory. *arXiv preprint arXiv:2602.21221*, 2026.

Vladislav Lialin, Vijeta Deshpande, Xiaowei Yao, and Anna Rumshisky. Scaling down to scale up: A guide to parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.15647*, 2023.

Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 1950–1965, 2022a.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pp. 100–114, 2022b.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. DoRA: Weight-decomposed low-rank adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32100–32121, 2024.

Yewei Liu, Xiyuan Wang, Yansheng Mao, Yoav Gelbery, Haggai Maron, and Muhan Zhang. SHINE: A scalable in-context hypernetwork for mapping context to LoRA in a single pass. *arXiv preprint arXiv:2602.06358*, 2026.

Arvind Mahankali, Tatsunori Hashimoto, and Tengyu Ma. One step of gradient descent is provably the optimal in-context learner with one layer of linear self-attention. In *International Conference on Learning Representations*, pp. 32077–32096, 2024.

Hanna Mazzawi, Benoit Dherin, Michael Munn, Adrian Goldwaser, Michael Wunder, and Javier Gonzalvo. Transmuting prompts into weights. *arXiv preprint arXiv:2510.08734*, 2025.

Maxime Meyer, Mario Michelessa, Caroline Chaux, and Vincent Y. F. Tan. Memory limitations of prompt tuning in transformers. *arXiv preprint arXiv:2509.00421*, 2025.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 11048–11064, 2022.

Ryumei Nakada, Wenlong Ji, Tianxi Cai, James Zou, and Linjun Zhang. A theoretical framework for prompt engineering: Approximating smooth functions with transformer prompts. *arXiv preprint arXiv:2503.20561*, 2025.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*, 2022.

Jane Pan, Tianyu Gao, Howard Chen, and Danqi Chen. What in-context learning "learns" in-context: Disentangling task recognition and task learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 8298–8319, 2023.

Aleksandar Petrov, Philip Torr, and Adel Bibi. Prompting a pretrained transformer can be a universal approximator. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 40523–40550, 2024a.

Aleksandar Petrov, Philip Torr, and Adel Bibi. When do prompting and prefix-tuning work? a theory of capabilities and limitations. In *International Conference on Learning Representations*, pp. 6031–6054, 2024b.

Eric Todd, Millicent Li, Arnab Sen Sharma, Aaron Mueller, Byron Wallace, and David Bau. Function vectors in large language models. In *International Conference on Learning Representations*, pp. 17282–17333, 2024.

Johannes von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 35151–35174, 2023.

Haiyu Wang and Yuanyuan Lin. Prompt tuning transformers for data memorization. In *Advances in Neural Information Processing Systems*, volume 38, pp. 27215–27256, 2025.

Haonan Wang, Brian K. Chen, Siquan Li, Xinhe Liang, Hwee Kuan Lee, Kenji Kawaguchi, and Tianyang Hu. PrefixMemory-Tuning: Modernizing prefix-tuning by decoupling the prefix from attention. In *International Conference on Learning Representations*, pp. 130687–130708, 2026.

Yihan Wang, Jatin Chauhan, Wei Wang, and Cho-Jui Hsieh. Universality and limitations of prompt tuning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 75623–75643, 2023.

Noam Wies, Yoav Levine, and Amnon Shashua. The learnability of in-context learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 36637–36651, 2023.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit Bayesian inference. In *International Conference on Learning Representations*, 2022.

Qingyu Yin, Xuzheng He, Chak Tou Leong, Fan Wang, Yanzhao Yan, Xiaoyu Shen, and Qiang Zhang. Deeper insights without updates: The power of in-context learning over fine-tuning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 4138–4151, 2024.

Yuchen Zeng and Kangwook Lee. The expressive power of low-rank adaptation. In *International Conference on Learning Representations*, pp. 5078–5123, 2024.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adaptive budget allocation for parameter-efficient fine-tuning. In *International Conference on Learning Representations*, 2023.

The appendices follow the main text. Appendix A collects the notation, and Appendix B extends the related work. Appendices C and D prove the results of Section 4, Appendices E and F those of Section 5, and Appendices G and H those of Section 6. Appendices I and J give the protocols behind Section 7.

A NOTATION

Table 3 lists the symbols used in the main text, grouped by the section that introduces them. One clash is deliberate and local: the bandwidth h of the signed construction (Theorem 11) appears only in Appendix G and Algorithm 2, whereas subscripted h_X and $h_{S,X}$ always denote head outputs.

Table 3: Notation, grouped by the part of the paper that introduces each symbol.

Symbol	Meaning
Attention interface	
$d,\ d_k,\ d_v,\ d_o$	input, key, value, and output dimensions
W^Q, W^K, W^V, W^O	frozen query, key, value, and output-projection weights
$X = (x_1, \dots, x_j)$	content states up to the readout position j
$q,\ k_i,\ v_i,\ s_i$	readout query, content keys and values, scaled logits
$Z_X,\ N_X,\ h_X$	partition function, value numerator, frozen head output
$S = \{(\kappa_t, \nu_t)\}_{t=1}^m$	independent key–value prefix with m slots
$Z_S(q),\ N_S(q)$	prefix mass and prefix numerator at query q
$h_{S,X}$	prefixed head output
$T = T_\Delta$	adapter target, the head output after the update Δ
$\Delta W^V = BA; B, A, r$	value LoRA factors and rank
$\mathcal{D},\ \mathcal{E}_\mathcal{D}(S,T)$	input domain and uniform prefix error
$\mathcal{P}_{\mathrm{KV}}^{<\infty}$	all finite nonempty independent key–value prefixes
Observability	
$\rho,\ g_S(q)$	prefix mixing weight and mean prefix value
$\Sigma(X) = (q, Z_X, N_X)$	content summary; equal Σ defines a Σ-fiber
$K,\ V$	per-slot caps on key and value norms
H	bound on frozen output norms $\|h_X\|_2$
$\omega(X, X')$	summary modulus of a pair
Realizability	
q_0	common (or reference) query
$a,\ b$	aggregate prefix mass and numerator at q_0
$Z_0,\ t,\ c$	common partition, contraction factor, translation
$L,\ \mathcal{A}_m$	key-cap logit range and attainable aggregate set
G	fixed linear map on head outputs, e.g. $G = W^O$
ϵ	tolerance of a feasibility test or construction
a_0	amplitude of a symmetric scalar target
$E_*^G,\ E_0^G,\ d_0$	capped optimum, reference optimum, query slack
$D_\mathcal{D}^G,\ e_*^G$	unprefixed adapter effect and normalized optimum
Construction and witnesses	
$\alpha,\ \eta_\alpha$	query-update scale and its minimax error floor
$u_0, \dots, u_r,\ s_0$	query-exposure vectors and constant self-score
$M,\ C_0$	bounds on adapter coordinates and frozen values
$\delta,\ h,\ \beta,\ \gamma$	mass fraction, bandwidth, logit intercept, value scale
e	query perturbation in the near-fiber experiment

B EXTENDED RELATED WORK

Implicit updates and context-to-weight conversion. In-context learning stores task information in activations supplied at inference time, whereas LoRA stores it in a parameter update (Brown et al., 2020; Hu et al., 2022). Linear and simplified constructions show that transformers can implement least-squares or gradient-based updates over demonstrations (Garg et al., 2022; Akyürek et al., 2023; von Oswald et al., 2023; Mahankali et al., 2024; Ahn et al., 2023), and statistical analyses study task selection, generalization, and sample complexity (Xie et al., 2022; Bai et al., 2023; Wies et al., 2023;

Li et al., 2023). Dai et al. (2023) interpret attention as a dual form of gradient descent. Several works convert context into weights: Chen et al. (2024) obtain an exact conversion for linearized attention by adding bias terms, Dherin et al. (2025) derive context-dependent low-rank updates to an MLP inside a transformer block, Mazzawi et al. (2025) aggregate prompt effects into reusable weight-space interventions, and Liu et al. (2026) map context to LoRA through a learned hypernetwork. Latent context compilation instead uses a temporary adapter to produce compact portable tokens (Li et al., 2026). All of these run from context to parameters; we study the reverse direction, whether a fixed adapter can be replaced by one nonadaptive prefix at a frozen head.

Prompt expressivity and memorization limits. Wang et al. (2026) identify the competition for attention mass implied by the prefix decomposition and move the prefix outside the attention head. Wang et al. (2023) give universality and finite-depth limitation results and compare prompt parameters with low-rank updates, and Meyer et al. (2025) show that the information a prompt can memorize grows at most linearly in its length. Further results extend universality or quantify memorization limits under other assumptions (Petrov et al., 2024a; Hu et al., 2025; Wang & Lin, 2025; Nakada et al., 2025; Hsu & Lai, 2026). These results concern model classes or datasets. Our tests instead fix one head, the prefix interface, and a declared adapter target, and the observability floor holds at every prefix length.

Parameter-efficient adaptation and task representations. LoRA constrains an update to a low-rank factorization whose expressive power has been characterized (Zeng & Lee, 2024), and it belongs to a broader family of parameter-efficient methods (Houlsby et al., 2019; Han et al., 2024; Liu et al., 2022a; Zhang et al., 2023; Liu et al., 2024; Lialin et al., 2023). Prompt-induced behavior can also be represented through task or function vectors (Hendel et al., 2023; Todd et al., 2024). Adaptive task vectors are input-dependent and are argued to match LoRA expressivity under a rank-matched construction (Kang et al., 2025); such an input-dependent interface lies outside our setting, which requires one prefix shared by all inputs. Induction-head analyses explain how repeated patterns are copied in context (Olsson et al., 2022; Elhage et al., 2021), and retrieval supplies factual information through context rather than parameters (Lewis et al., 2020). Empirically, neither in-context learning nor fine-tuning dominates: demonstration labels, ordering, and format matter (Min et al., 2022; Liu et al., 2022b; Pan et al., 2023), parameter-efficient tuning can be cheaper and stronger in few-shot settings (Liu et al., 2022a), and other controlled tasks favor in-context generalization (Yin et al., 2024). Dong et al. (2024) survey the wider field.

C Proofs of observability results

C.1 Prefix decomposition and completeness

Splitting (3) into its content and prefix contributions gives

$$\frac{N_X}{Z_X + Z_S} + \frac{N_S}{Z_X + Z_S} = \frac{Z_X}{Z_X + Z_S} h_X + \frac{Z_S}{Z_X + Z_S} g_S(q),$$

which proves Lemma 1. The only content-dependent arguments are q, Z_X, N_X.

Suppose all one-slot interventions have identical outputs on two inputs. For the zero key, the exponential mass is one, so

$$\frac{N + \nu}{Z + 1} = \frac{N' + \nu}{Z' + 1} \quad \text{for every } \nu \in \mathbb{R}^{d_v}.$$

Equality of the coefficients of ν gives $Z = Z'$, and the constant terms give $N = N'$. For an arbitrary key κ, let $a = \exp(q^\top \kappa / \sqrt{d_k})$, and define a' analogously. The coefficients of ν now give $a/(Z + a) = a'/(Z + a')$. Strict monotonicity in the positive argument implies $a = a'$. Taking logarithms yields $(q - q')^\top \kappa = 0$ for every κ, hence $q = q'$. The converse follows immediately from (3).

C.2 Fiber diameter and relative attention

For an equal-summary pair, write its common prefixed output as y_S. Then

$$\|T(X) - T(X')\|_2 \le \|T(X) - y_S\|_2 + \|y_S - T(X')\|_2 \le 2\mathcal{E}_{\mathcal{D}}(S, T).$$

Taking the supremum over pairs and the infimum over prefixes proves Proposition 3. The factor one-half is exact for the symmetric scalar witness in Theorem 9. If the supremum is zero, the proposition gives only a zero lower bound; it makes no positive realizability claim.

The same denominator cancellation recovers the established relative-content attention invariant (Petrov et al., 2024b). For any two content indices $i, \ell \le j$,

$$\frac{a^S_{ji}}{a^S_{j\ell}} = \frac{e^{s_i}/(Z_X + Z_S)}{e^{s_\ell}/(Z_X + Z_S)} = e^{s_i - s_\ell} = \frac{a_{ji}}{a_{j\ell}}. \tag{21}$$

An invariant attention ratio is not by itself an invariant output because prefix values can compensate. The common-summary argument removes that ambiguity by equating every quantity entering the response.

D Proof of the near-fiber bound

We prove Theorem 4 by bounding, uniformly over capped prefixes, how each factor of the mixture (5) moves with the summary.

Let $w_t(q)$ be the softmax weights on prefix slots alone. The function $\log Z_S(q)$ has gradient $\sum_t w_t \kappa_t/\sqrt{d_k}$, whose norm is at most $K/\sqrt{d_k}$. The mean prefix value is $g_S(q) = \sum_t w_t \nu_t$, so $\|g_S(q)\| \le V$. For unit vectors $u \in \mathbb{R}^{d_v}$ and $v \in \mathbb{R}^{d_k}$,

$$u^\top (Dg_S(q))v = \frac{1}{\sqrt{d_k}} \operatorname{Cov}_{w(q)}(u^\top \nu_t, v^\top \kappa_t).$$

Cauchy–Schwarz and $\operatorname{Var}(u^\top \nu_t) \le \mathbb{E}(u^\top \nu_t)^2 \le V^2$, with the analogous bound K^2, imply $\|Dg_S(q)\|_{\mathrm{op}} \le VK/\sqrt{d_k}$. Integration along the segment between queries makes g_S Lipschitz with this constant, independent of m.

Write $\rho = \sigma(\log Z_S(q) - \log Z_X)$, where σ is the logistic sigmoid. Since $\sup|\sigma'| = 1/4$,

$$|\rho - \rho'| \le \tfrac{1}{4}\left(\frac{K}{\sqrt{d_k}}\|q - q'\|_2 + |\log Z_X - \log Z_{X'}|\right).$$

Using $y = (1-\rho)h + \rho g$ and the primed analogue,

$$y - y' = (1-\rho)(h - h') + \rho(g - g') + (\rho - \rho')(g' - h').$$

The triangle inequality, $0 \le \rho \le 1$, and $\|g' - h'\| \le V + H$ prove (8). Finally,

$$\|T - T'\| \le \|T - y\| + \|y - y'\| + \|y' - T'\| \le 2\max\{\|T - y\|, \|T' - y'\|\} + \omega.$$

Rearrangement and nonnegativity prove (9). Exact summary equality sets $\omega = 0$ and recovers the original pairwise fiber bound. Without bounds on keys and values, the derivative argument supplies no uniform modulus over prefixes.

E Proofs of exact realizability results

This appendix proves Theorem 5 and the two-input and bounded-key statements behind Corollary 7.

E.1 Common query and partition

For a common query $q_0 \neq 0$, any prefix determines two shared quantities $a = Z_S(q_0) > 0$ and $b = N_S(q_0)$. Thus its responses have the form on the right of (10). Conversely, the one-slot key $\sqrt{d_k}\log(a) q_0/\|q_0\|^2$ has mass a, and value b/a has numerator b. The two infima are identical.

For a fixed tolerance ϵ, multiplying by the positive denominator $Z_X + a$ gives the stated second-order-cone inequalities. Their left sides are norms of affine functions of (a, b) and their right sides are positive affine functions on $a > 0$. Sublevel feasibility is therefore convex. The strict positivity restriction matters for infima at $a \downarrow 0$; numerical implementations can use positive lower bounds and inspect limits rather than claiming that an unattained boundary point is a finite prefix.

If $Z_X = Z_0$, every prefix has $t = Z_0/(Z_0 + a) \in (0, 1)$ and $c = b/(Z_0 + a)$. Every such (t, c) is realized because $a = Z_0(1 - t)/t$ and $b = cZ_0/t$. On the finite domain, or a bounded frozen-output domain, $t \downarrow 0$ and $t \uparrow 1$ give uniform limits with arbitrary fixed c. These are precisely the additional maps in the closure. If $q_0 = 0$, all slot logits are zero, so $Z_S = m$ and this continuous mass characterization does not hold.

E.2 Two-point minimax formula

For fixed t, the two residuals before translation are $e^{\pm} = T(X^{\pm}) - th_{X^{\pm}}$. Any translation c incurs maximum error at least $\|e^+ - e^-\|/2$ by the triangle inequality. Choosing $c = (e^+ + e^-)/2$ attains this value. Minimizing over the compact interval $[0, 1]$ gives (12). Endpoint optima are interpreted through the prefix closure and need not be attained by finite parameters.

For the matched updates, $T^{\pm} = \pm a_0$ and $h^{\pm} = \pm 1$. The minimax formula reduces to $\min_{0 \le t \le 1} |a_0 - t|$. It equals zero at $a_0 = .5$ and $.5$ at $a_0 = 1.5$. Each matrix update is a nonzero 1×2 row, hence rank one; its Frobenius norm is $.5$. The unprefixed maximum error is also $.5$ in both cases. Since $N = \pm 1$, the two summaries differ, so no nonidentical equal-summary pair contributes to (7).

With m slots and scalar keys in $[-K, K]$, the prefix mass belongs to $[me^{-K}, me^{K}]$. Every intermediate mass is attained using the common key $\log(a/m)$. Therefore the contraction interval in (16) is exact. For a symmetric target, the optimal translation is zero, and zero values realize it. The optimal t is the projection of a_0 onto that interval. This proves both the bounded optimum and the construction used in the experiment.

F Capped feasibility, reconstruction, and reference-query bounds

This appendix proves Theorem 6, turns it into a numerically safe test, and proves Proposition 8.

F.1 Exact attainable aggregates

For $q_0 \neq 0$, Cauchy–Schwarz gives $|q_0^\top \kappa_t|/\sqrt{d_k} \le L$. Summing the exponential masses yields the interval in (13). The triangle inequality gives

$$\|b\|_2 = \left\| \sum_t e^{q_0^\top \kappa_t/\sqrt{d_k}} \nu_t \right\|_2 \le V \sum_t e^{q_0^\top \kappa_t/\sqrt{d_k}} = Va.$$

Conversely, for any $(a, b) \in \mathcal{A}_m$, Equation (15) has

$$\|\kappa_t\|_2 = \frac{\sqrt{d_k}|\log(a/m)|}{\|q_0\|_2} \le K, \qquad \|\nu_t\|_2 = \|b\|_2/a \le V.$$

Its per-slot mass is exactly a/m, so summing gives $(Z_S, N_S) = (a, b)$. When $q_0 = 0$, every slot mass is one regardless of its key; thus $a = m$, and zero keys with $\nu_t = b/m$ realize the full ball $\|b\| \le Vm$. These arguments include $K = 0$ and $V = 0$.

The set $\mathcal{A}_m$ is nonempty, convex, and compact. Each $Z_X > 0$, so the objective

$$\max_X \left\| G\left[\frac{N_X + b}{Z_X + a} - T(X)\right] \right\|_2$$

is continuous on that set and attains its minimum. At fixed ϵ, multiplication by the positive denominator gives (14); its left side is a norm of an affine function and its right side is positive and affine. The cap $\|b\| \le Va$ is also a second-order-cone constraint. This proves Theorem 6. The linear map G may be rank deficient. Nullspace directions disappear from the error objective but not from the original value cap.

F.2 A numerically explicit feasibility procedure

The zero-key case in the initialization line uses $L = 0$. The displayed optimum is a mathematical characterization. Floating-point solver status alone is not an exact lower-bound certificate: a rigorous

Algorithm 1 Capped finite-domain prefix test

Require: finite summaries (q_0, Z_i, N_i), targets T_i, slots m, caps K, V, map G, tolerance $\xi > 0$

define $\mathcal{A}_m$ using (13), or $a = m$ if $q_0 = 0$
choose $a_0 = me^{-L}$, $b_0 = 0$; retain $(a, b) = (a_0, b_0)$ and set $\epsilon_{\mathrm{lo}} = 0$
set $\epsilon_{\mathrm{hi}} = \max_i \|G[N_i/(Z_i + a_0) - T_i]\|_2$
while $\epsilon_{\mathrm{hi}} - \epsilon_{\mathrm{lo}} > \xi$ **do**
    $\epsilon \leftarrow (\epsilon_{\mathrm{hi}} + \epsilon_{\mathrm{lo}})/2$
    solve feasibility of (14) at ϵ
    if a feasible pair is verified by direct residual evaluation **then**
        retain (a, b); tighten ϵ_{hi} using its directly evaluated maximum error
    else if a validated infeasibility certificate is obtained **then**
        $\epsilon_{\mathrm{lo}} \leftarrow \epsilon$
    else
        stop and return the unresolved interval, not an impossibility conclusion
    end if
end while
reconstruct the feasible prefix by (15); evaluate its actual attention error

numerical exclusion needs a dual infeasibility certificate or validated residual bounds. Report primal feasibility residuals, the bisection interval, solver tolerances, and any unresolved cells rather than treating a solver failure as infeasibility.

A common numerical rescaling of all Z_i, N_i, a, b leaves the responses unchanged and must also rescale the mass interval. Independently shifting the logits for different examples rescales their content summaries differently and does not preserve a single common (a, b); it cannot be done without tracking those example-specific factors in the constraints. For a count budget of at most m, solve each count $k = 1, \ldots, m$ and take the best value. When permitted, the zero-slot option is the frozen output. The minimum over these nested count sets is nonincreasing; any deterioration reported for exactly m slots does not contradict this monotonicity.

F.3 Compactifying the unconstrained common-query infimum

For Theorem 5 without caps, choose an arbitrary reference mass $Z_{\mathrm{ref}} > 0$ and define

$$t = \frac{Z_{\mathrm{ref}}}{Z_{\mathrm{ref}} + a} \in (0, 1), \qquad c = \frac{b}{Z_{\mathrm{ref}} + a}, \qquad d_i(t) = 1 + t\left(\frac{Z_i}{Z_{\mathrm{ref}}} - 1\right).$$

Then

$$\frac{N_i + b}{Z_i + a} = \frac{tN_i/Z_{\mathrm{ref}} + c}{d_i(t)}.$$

For $0 \le t \le 1$, $d_i(t) \ge \min\{1, Z_i/Z_{\mathrm{ref}}\} > 0$. Closing the interval to $[0, 1]$ therefore includes the limiting constant maps at $a \to \infty$ and the finite-output limits at $a \downarrow 0$. The fixed-tolerance constraints become

$$\|G\,[tN_i/Z_{\mathrm{ref}} + c - d_i(t)T_i]\|_2 \le \epsilon d_i(t), \qquad 0 \le t \le 1,$$

which are again second-order-cone constraints. Interior solutions reconstruct a finite prefix; endpoint solutions may describe only an infimum. For $G = I$, one input's bounded residual bounds c on every objective sublevel set. For a rank-deficient G, restrict c to the orthogonal complement of $\ker G$ without changing projected responses; on this space the same compactness argument applies. Thus the closed formulation attains the projected infimum and records whether a finite-parameter realization was actually found.

F.4 Proof of the reference-query sandwich

Define the algebraic response $y_S(q, Z, N)$ by (3), even for a summary not realized by a content sequence. For fixed $h = N/Z$, changing only q gives

$$\|y_S(q, Z, N) - y_S(q_0, Z, N)\|_2 \le \left(V + \frac{H + V}{4}\right) \frac{K}{\sqrt{d_k}} \|q - q_0\|_2,$$

using the prefix-mean Lipschitz constant and sigmoid derivative from Appendix D. The bound holds for every prefix satisfying the caps. Consequently, its maximum target errors on the original and reference summaries differ by at most $\|G\|_{\mathrm{op}} d_0$. Taking infima gives both sides of (18). A prefix realizing the reference optimum obeys the same uniform inequality, which proves the upper-bound guarantee after reconstruction. No assumption that the reference summaries arise from real activations is used.

This argument also supplies a sharper feasibility exclusion when query distances vary substantially. For a candidate tolerance ϵ on the original domain, replace the common tolerance on input i by $\epsilon + \|G\|_{\mathrm{op}} d_i$, where

$$d_i = \left(V + \frac{\|h_i\|_2 + V}{4} \right) \frac{K}{\sqrt{d_k}} \|q_i - q_0\|_2.$$

Any feasible original prefix induces a feasible aggregate pair for these relaxed constraints. Their infeasibility therefore rules out the original tolerance, whereas feasibility alone does not prove it. The relaxation remains an SOCP at fixed ϵ.

F.5 First-layer applicability and experimental separation

The pinned GPT-2 implementation forms input states from token and positional embeddings, applies deterministic evaluation-mode dropout and tokenwise pre-attention layer normalization, and computes its first-block affine projections (Hugging Face, 2024). Fixing a readout token and its explicit position therefore fixes its query regardless of preceding token identities. Projection biases are fixed and do not affect this equality. Cached content keys, values, and masks still depend on the content. A structural row slice of a fused query–key–value projection must be verified before labeling an adapter query-only or value-only.

For a single changed head, $G = W^O$ evaluates that head's contribution to the residual stream with the other contributions fixed. A positive local optimum can be erased downstream and does not imply a token-level impossibility. The executable evaluation in Appendix J specifies the saved checkpoint and adapter states, domain construction, separate fitting and evaluation inputs, query checks, and solver outputs. Its unexecuted status is distinct from the controlled results in Section 7.

G Proof of the signed value construction

The contraction obstruction does not preclude compilation when the query exposes the coordinates needed for a correction. The following assumption makes that exposure explicit.

Assumption 10 (Affine query exposure)**.** Let $\mathcal{K} \subset \mathbb{R}^d$ be nonempty and compact, with $z(x) = Ax \in [-M, M]^r$. At the one-token input (x), the self-score is constant s_0. There are $u_0, u_1, \ldots, u_r \in \mathbb{R}^{d_k}$ such that $u_0^\top W^Q x = 1$ and $u_i^\top W^Q x = (Ax)_i$ on $\mathcal{K}$. Also $\sup_{x \in \mathcal{K}} \left\| W^V x \right\|_2 \le C_0$.

A concrete instance is $x = (1, z)$, $W^Q = I$, $W^K = 0$, and A selecting z. The assumption is not automatic at a pretrained head and does not hold merely because the adapter is value-side.

Theorem 11 (Two slots per exposed value direction)**.** *Under Assumption 10, every declared update $\Delta W^V = BA$ can be approximated uniformly on $\mathcal{K}$ to any $\epsilon > 0$ by one independent-KV prefix with exactly $2r$ slots. For $\delta, h > 0$, take*

$$\beta = \log \frac{\delta e^{s_0}}{2r}, \qquad \gamma = \frac{r(1+\delta)}{\delta h},$$
$$\kappa_i^{\pm} = \sqrt{d_k}(\beta u_0 \pm h u_i), \qquad \nu_i^{\pm} = \pm \gamma B_{:i}. \tag{22}$$

When $hM \le 1$, its uniform error is bounded by

$$\mathcal{E} \le \delta \cosh(1) C_0 + \delta(\cosh(1) - 1) \|B\|_{\mathrm{op}} \sqrt{r} M + (1+\delta) \|B\|_{\mathrm{op}} \frac{\sqrt{r}\, h^2 M^3 \cosh(1)}{6}. \tag{23}$$

Choosing $\delta = \Theta(\epsilon)$ and $h = \Theta(\sqrt{\epsilon})$ gives $\mathcal{E} \le \epsilon$ with $\gamma = O(\epsilon^{-3/2})$.

Algorithm 2 Constructing a signed prefix under verified exposure

Require: factors B, A, exposure vectors $u_0, \dots, u_r$, constant self-score s_0, bounds C_0, M, tolerance $\epsilon > 0$

choose $\delta > 0$ so the two attenuation terms of (23) are at most $\epsilon/2$
choose $h > 0$ with $hM \le 1$ so its Taylor term is at most $\epsilon/2$
$\beta \leftarrow \log(\delta e^{s_0}/(2r))$, $\gamma \leftarrow r(1+\delta)/(\delta h)$
for $i = 1, \dots, r$ **do**
    $\kappa_i^{\pm} \leftarrow \sqrt{d_k}(\beta u_0 \pm h u_i)$
    $\nu_i^{\pm} \leftarrow \pm\gamma B_{:i}$
end for
return the $2r$ independent key–value pairs

The signed pairs produce $\sinh(hz_i)/h = z_i + O(h^2)$, while a small total prefix mass limits attenuation of the base output. Their values compensate for both the small mass and bandwidth. This establishes a sufficient rank-dependent construction, not a minimal slot count or a universal lower bound on parameter norms. Its precision cost is measured directly in Section 7.

Write $z = Ax$, $b_i = B_{:i}$, $E = e^{s_0}$, and $a = e^{\beta}$. Query exposure gives the signed pair logits $\beta \pm hz_i$. With values $\pm\gamma b_i$, their total numerator is $2a\gamma \sum_i \sinh(hz_i) b_i$. Let $D_0 = E + 2ar$ and choose $\gamma = D_0/(2ah)$. The full denominator and output are

$$D(z) = E + 2a \sum_i \cosh(hz_i),$$
$$h_{S,(x)} = \frac{E}{D(z)} W^V x + \frac{D_0}{D(z)} B\widetilde{z}, \qquad \widetilde{z}_i = \frac{\sinh(hz_i)}{h}. \tag{24}$$

The target is $W^V x + Bz$. Set $\delta = 2ar/E$. For $hM \le 1$,

$$0 \le D(z) - E \le E\delta \cosh(1),$$
$$|D(z) - D_0| \le E\delta(\cosh(1) - 1),$$
$$\|\widetilde{z} - z\|_2 \le \frac{\sqrt{r}\, h^2 M^3 \cosh(1)}{6}.$$

The last inequality is the third-order Taylor remainder for $\sinh$. Subtracting the target in (24), using $D(z) \ge E$, and bounding $\|Bz\| \le \|B\|_{\mathrm{op}} \sqrt{r} M$ give (23). Choose δ to make the two attenuation terms at most $\epsilon/2$, then choose positive h so that $hM \le 1$ and the Taylor term is at most $\epsilon/2$. This proves uniform approximation using exactly $2r$ slots. If a coefficient vanishes, its corresponding restriction can simply be omitted; the zero-update case does not create an obstruction.

Finally, $a = \delta E/(2r)$ gives $\beta = \log(\delta E/(2r))$ and $\gamma = r(1+\delta)/(\delta h)$. A schedule $\delta = \Theta(\epsilon)$, $h = \Theta(\sqrt{\epsilon})$ has $\gamma = O(\epsilon^{-3/2})$ and a logarithmically diverging logit-intercept magnitude. The bound is an exact-real-arithmetic statement. It does not control cancellation and rounding after keys and values are represented in finite precision.

H Witness matrices and output projections

This appendix gives the witness matrices behind Theorem 9 and the value counterexample, and carries both through the output projection.

The same-head witness has

$$W^Q = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \qquad W^K = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \qquad W^V = [0 \quad 1 \quad 0].$$

At $x_3 = e_3$, $q = e_1$. The first two keys are $\pm e_2$, so every frozen logit is zero and their values are $(1, -1, 0)$. Swapping the keys while retaining the two values gives the two inputs in the main text; both have $Z = 3, N = 0$.

For $\Delta W^Q = \sqrt{2}\alpha e_2 e_3^\top$, the query becomes $e_1 + \sqrt{2}\alpha e_2$. Scaled logits are $(\alpha, -\alpha, 0)$ on X^+ and $(-\alpha, \alpha, 0)$ on X^-. The targets are therefore

$$\pm\frac{e^\alpha - e^{-\alpha}}{e^\alpha + e^{-\alpha} + 1} = \pm\eta_\alpha.$$

Every shared prefix has one common output, so its maximum error is at least η_α. A prefix with zero values has numerator zero on both inputs and attains the bound. It is thus an exact minimax value, not only a lower bound.

For $\Delta W^V = ce_3^\top$, values become $(1, -1, c)$ under the unchanged uniform attention. Both target outputs equal $c/3$. A single zero-key slot with value $4c/3$ gives output $(4c/3)/(3+1) = c/3$, proving exact value compilation on the same head and domain. Both updates are rank one for their nonzero scales.

Proposition 12 (Value-side observability failure)**.** *With input dimension two, scalar keys and values, $W^Q = W^K = W^V = 0$, and $X^\pm = (\pm e_1, e_2)$, both summaries equal $(0, 2, 0)$. The rank-one update $\Delta W^V = [1\ 0]$ gives targets $\pm 1/2$. Every shared prefix has worst-case error at least $1/2$.*

Uniform content attention gives the two targets directly, and Proposition 3 proves the result. This is an observability failure; the amplifying example in Corollary 7 is instead a realizability failure with an observable target.

Corollary 13 (Output projection)**.** *An approximation error at most ϵ before W^O becomes at most $\left\|W^O\right\|_{\mathrm{op}} \epsilon$ after projection. In the scalar query witness, let $w = W^O(1)$. Its projected minimax error is exactly $\eta_\alpha \left\|w\right\|_2$.*

The upper bound is submultiplicativity. Every prefix gives the pair a common scalar y, whose projected target errors are $(y - \eta_\alpha)w$ and $(y + \eta_\alpha)w$. Their maximum norm is at least $\eta_\alpha \left\|w\right\|$, and $y = 0$ attains it. The obstruction disappears precisely when $w = 0$. A deeper network may also change or erase the witness, so a head-level lower bound is not automatically an output-token lower bound.

I Controlled experimental protocols

Each protocol below specifies one controlled experiment of Section 7.

I.1 Direct construction and numerical precision

For rank $r \in \{1, 2, 4, 8\}$, the input is $x = (1, z) \in \mathbb{R}^{r+1}$, query projection is the identity, key projection is zero, and value dimension is $r + 2$. The frozen value map sends the constant coordinate to a random unit vector v_0 and every varying coordinate to zero. Thus $W^V x = v_0$, $C_0 = 1$, self-score $s_0 = 0$, and $M = 1$. A random $(r + 2) \times r$ matrix is normalized to operator norm one to obtain B. The update matrix A selects the last r input coordinates. This construction verifies the exposure assumption rather than estimating it.

For each of 20 seeds, we draw 10,000 independent uniform cube points and append all 2^r vertices. The same head and test set are used for all tolerances and precisions. Let

$$c_r = \cosh(1) + (\cosh(1) - 1)\sqrt{r}, \quad \delta = \frac{\epsilon}{2c_r}, \quad h = \min\left\{1, \sqrt{\frac{3\epsilon}{(1+\delta)\sqrt{r}\cosh(1)}}\right\}.$$

Together with (22), these parameters make the analytic error bound at most ϵ. The tolerances are $10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-6}$.

Direct attention is evaluated in float64, float32, and bfloat16. The attention temperature is absorbed into the stored keys, so the represented key parameters are $\kappa/\sqrt{d_k}$; inputs, these scaled keys, and values are cast to the designated precision before the matrix products and softmax. Errors are computed against the high-precision target. A stable float64 evaluation of (24) is also recorded as a construction check. The experiment has $4 \times 20 \times 5 \times 3 = 1200$ numerical rows. Passing means the finite test-set maximum does not exceed the requested tolerance; the exact-arithmetic uniform guarantee comes from the theorem, not from this test.

Table 4: Tolerance passes among 100 seed–tolerance cases per rank and precision. Every precision uses the same analytic construction and test points.

Rank	Float64	Float32	Bfloat16
1	100/100	80/100	20/100
2	100/100	80/100	17/100
4	100/100	69/100	1/100
8	100/100	60/100	0/100

Table 5: Exact best worst-case error on the matched two-input domain, with per-slot key cap $K = 4$. Both unprefixed errors equal 0.5. The constructed prefix attains the bounded optimum; no optimizer performance is involved.

Slots	Contracting target $a_0 = .5$	Amplifying target $a_0 = 1.5$
1	0.0000	0.5180
4	0.0000	0.5683
16	0.0000	0.7266
64	0.0396	1.0396

The rank-two float64 mean sampled maximum at tolerance 10^{-2} is 0.00439 with sample SD 0.00038; at 10^{-6} it is 4.40×10^{-7} with SD 3.77×10^{-8}. Bfloat16 at tolerance 10^{-3} has mean sampled maximum 1.1029 and SD 0.0676. These spreads vary random heads and test sets through the seed; they are not standard errors or estimates of worst-case failure probability. The complete numerical output additionally records mean and 99.9th-percentile error, the analytic bound, bandwidth, mass, logit intercept, value scale, and maximum value norm.

I.2 Matched-effect realizability

Use the two one-token inputs $x^{\pm} = (1, \pm 1)$, scalar query $q = 1$, zero content keys, and frozen output $h^{\pm} = \pm 1$. Targets are $T^{\pm} = \pm a_0$ with $a_0 = .5$ or 1.5. For key caps $K \in \{2, 4, 8\}$ and lengths $m \in \{1, 4, 16, 64\}$, let t_* be the projection of a_0 onto the interval (16). Give every slot the key

$$\kappa = \log \frac{1/t_* - 1}{m}$$

and value zero. The attention output is exactly $\pm t_*$, and its maximum error is $|t_* - a_0|$. This is both an attaining construction and the analytically optimal bounded-prefix value. The code verifies the cap and formula agreement for all 24 cases; these deterministic cases have no training-seed uncertainty.

I.3 Near-fiber optimization

Extend the query witness to input dimension four. The matrices satisfy

$$W^Q e_3 = e_1, \quad W^Q e_4 = e_2, \quad W^Q e_1 = W^Q e_2 = 0, \qquad W^K = e_2 e_1^\top, \quad W^V = e_2^\top.$$

Use

$$X^+ = (e_1 + e_2, -e_1 - e_2, e_3 + e e_4), \qquad X^- = (-e_1 + e_2, e_1 - e_2, e_3 - e e_4).$$

Then $q^{\pm} = e_1 \pm e e_2$, $Z^{\pm} = 2\cosh(e/\sqrt{2}) + 1$, and $N^{\pm} = 2\sinh(e/\sqrt{2})$. The query update is $\Delta W^Q = 2\sqrt{2} e_2 e_3^\top$, giving the targets stated in the main text. For $e \in \{0, .001, .01, .1\}$, $\|h\| \le 1$, and the two frozen outputs and log partitions agree. Equation (8) becomes

$$\omega = \left(1 + \frac{1+1}{4}\right) \frac{2}{\sqrt{2}} (2e).$$

The lower bound is computed on the exact two-input domain, with no nearest-neighbor approximation.

For each perturbation, a shared 16-slot prefix is initialized ten times with independent Gaussian keys and scalar values scaled by 0.2. Projected Adam uses learning rate 0.02 and 1000 steps. After each

Table 6: Interface and resource inputs for the first-layer GPT-2 evaluation. d_k is head width, d_o is residual width, and K, V are physical per-slot norm caps. Fitting and evaluation contexts are distinct design partitions.

Head	d_k	d_o	Fit	Evaluation	m	K	V
0	64	768	128	128	4	8.40	6.10
4	64	768	128	128	4	10.20	7.30
8	64	768	128	128	4	9.10	6.70

step, keys are projected to the radius-two Euclidean ball and values clipped to $[-1, 1]$. The objective is the maximum absolute target error across the pair; the best iterate is retained. Reporting its mean and SD across initializations characterizes the optimizer on a fixed problem, not performance across independent datasets. The smallest attained error is an upper bound on the bounded-prefix optimum, while the analytic expression is a lower bound. A gap between them may reflect either looseness of the theorem or optimization error.

J Direct GPT-2 numerical evaluation scenarios

Adapter effects, conic brackets, optimizer errors, and transfer errors are assumed; their normalizations and order relationships are calculated.

J.1 Interface, target, and resource specification

The design fixes the pretrained GPT-2 first attention block, the readout token, and its explicit position. It uses heads $0, 4, 8$, adapter ranks $1, 4$, and training seeds $0, 1, 2$. Fitting and evaluation domains each contain 128 contexts: 31 content token IDs followed by the same end-of-text token at position 31. Position IDs remain $0, \ldots, 31$, dropout is disabled, and cached states follow the first tokenwise layer normalization.

A value adapter changes only the selected head's value slice, and a query adapter changes only its query slice. The frozen affine projection biases remain in the computation. For value targets, frozen queries, keys, and attention weights remain unchanged; for query targets, frozen keys and values remain unchanged. The selected pretrained output block is $G = W^O$, with contributions of other heads held fixed. At this readout, the realized query displacement BAx_j, rather than nominal rank alone, determines the query target. The numerical scenarios do not substitute a random update for an unobserved trained adapter.

Table 6 supplies explicit cap values for the scenarios. These physical key and value caps are assumptions, not observed maxima of pretrained activations. All principal rows use exactly four slots. Content summaries and reconstructed physical keys use one common logit offset; independently normalized examples are not assigned an unchanged common prefix mass. A nonzero computed query discrepancy requires the reference-query slack rather than an unqualified exact-common-query conclusion.

J.2 Adapter effect and finite-domain approximation

Table 7 reports a complete numeric scenario for each head, target placement, and rank. D^G is the maximum projected adapter effect. The assumed conic bracket is $[E_{\mathrm{lo}}, E_{\mathrm{hi}}]$, and the normalized upper endpoint is E_{hi}/D^G. Every learned-prefix value exceeds the corresponding feasible upper endpoint; no failed optimization is treated as a lower bound. A physical reconstruction attaining the upper endpoint would be obtained from Equation (15), but its pretrained attention evaluation is not asserted by the assumed table.

These scenarios deliberately include both orderings of value and query targets. At head 0 and rank 1, the assumed residual fractions are .184 and .631; at head 4, they are .683 and .267. Such a pattern would be compatible with a target-specific boundary. Absolute effects remain visible so a small residual is not confused with a nearly zero adapter.

Table 7: Pretrained-head comparison. All errors use the selected output block $G = W^O$ and exactly 4 slots. The lower and upper conic endpoints differ by at most 10^{-6}; the last column divides the upper endpoint by the adapter effect D^G.

Head	Target	r	D^G	E_{lo}	E_{hi}	Learned best	E_{hi}/D^G
0	Value	1	0.0826	0.015198	0.015198	0.017263	0.184
0	Value	4	0.1462	0.059503	0.059503	0.063889	0.407
0	Query	1	0.0918	0.057925	0.057926	0.061139	0.631
0	Query	4	0.1584	0.085852	0.085853	0.092189	0.542
4	Value	1	0.0735	0.050200	0.050201	0.052038	0.683
4	Value	4	0.1327	0.098463	0.098463	0.102444	0.742
4	Query	1	0.0841	0.022454	0.022455	0.025398	0.267
4	Query	4	0.1439	0.045903	0.045904	0.051660	0.319
8	Value	1	0.1068	0.041438	0.041438	0.044108	0.388
8	Value	4	0.1813	0.102615	0.102616	0.108055	0.566
8	Query	1	0.0987	0.044513	0.044514	0.047968	0.451
8	Query	4	0.1724	0.083958	0.083959	0.090855	0.487

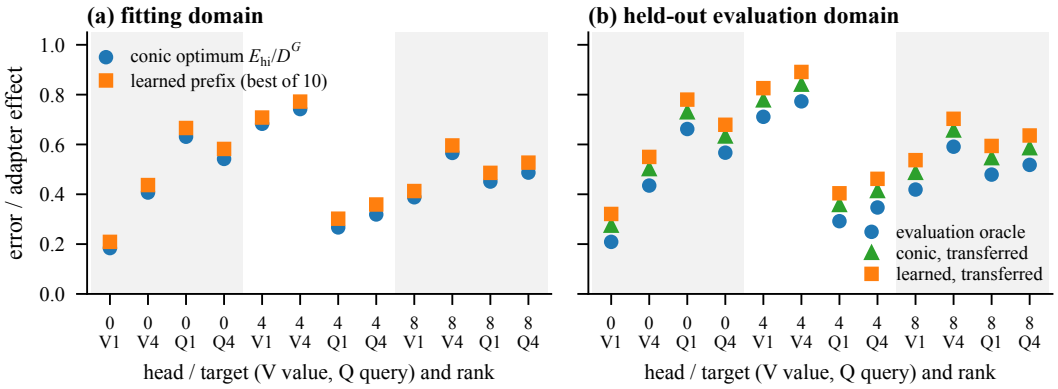


Figure 4: First-layer GPT-2 heads under the capped test with four slots and $G = W^O$. Each column is a head (0, 4, 8), target (V value, Q query), and rank. (a) Conic optimum and best learned prefix on the fitting domain, divided by the adapter effect D^G. (b) Worst-case error on held-out evaluation contexts, divided by the evaluation-domain effect.

J.3 Learned-prefix restart comparison and transfer

The gradient comparator uses ten independent restarts, 2000 projected-Adam steps per restart, and learning rate .01. Its objective is the maximum projected error on the same fitting domain and with the same caps as the conic problem. A conic-initialized run checks implementation consistency but does not replace the independent-restart comparator. The scenario's best and median restart errors are numerically distinct, so optimization quality is not summarized by an assumed conic result alone.

The evaluation-domain oracle and fitting-to-evaluation transfer answer different questions. Table 9 keeps a fixed evaluation normalization and gives the evaluation oracle a lower error than either transferred prefix. The conic prefix in the transfer column is fitted only on fitting targets, then frozen. Its transfer error is not the evaluation-domain optimum. The learned-transfer column follows the same separation.

J.4 Norm caps and the meaning of a slot budget

Table 10 varies both physical caps by the same multiplier. Each row is a fixed-count problem; an at-most budget also permits all smaller counts and the unprefixed output. The interior optima remain equal across counts when the aggregate mass intervals overlap, rather than assigning a benefit to extra slots by default. At the tightest caps, the exactly-16-slot scenarios instead incur additional attenuation. The at-most column takes the best permitted count and never increases with the budget.

Table 8: Same-domain optimizer comparison. The relative gap is $(E_{\mathrm{best}} - E_{\mathrm{hi}})/D^G$. Best and median summarize the ten restarts; no uncertainty interval or independent-dataset claim is attached to them.

Head	Target	r	Conic upper	Learned best	Learned median	Relative gap
0	Value	1	0.015198	0.017263	0.020485	0.025
0	Value	4	0.059503	0.063889	0.069737	0.030
0	Query	1	0.057926	0.061139	0.064903	0.035
0	Query	4	0.085853	0.092189	0.095990	0.040
4	Value	1	0.050201	0.052038	0.055346	0.025
4	Value	4	0.098463	0.102444	0.108549	0.030
4	Query	1	0.022455	0.025398	0.027837	0.035
4	Query	4	0.045904	0.051660	0.055977	0.040
8	Value	1	0.041438	0.044108	0.049555	0.025
8	Value	4	0.102616	0.108055	0.114219	0.030
8	Query	1	0.044514	0.047968	0.051423	0.035
8	Query	4	0.083959	0.090855	0.097061	0.040

Table 9: Held-out transfer comparison. All values are worst-case projected errors divided by the evaluation-domain adapter effect. The oracle uses evaluation targets; the transferred prefixes do not. These ratios are not a generalization guarantee.

Head	Target	r	Evaluation oracle	Conic transfer	Learned transfer
0	Value	1	0.209	0.274	0.321
0	Value	4	0.435	0.501	0.550
0	Query	1	0.662	0.729	0.780
0	Query	4	0.567	0.632	0.679
4	Value	1	0.711	0.777	0.826
4	Value	4	0.773	0.840	0.891
4	Query	1	0.292	0.357	0.404
4	Query	4	0.347	0.413	0.462
8	Value	1	0.419	0.486	0.537
8	Value	4	0.591	0.656	0.703
8	Query	1	0.479	0.545	0.594
8	Query	4	0.518	0.585	0.636

J.5 Numerical validation requirements

The conic feasibility check uses the tolerance and accounting in Algorithm 1. Primal cap and response residuals determine a feasible reconstructed upper endpoint; a lower endpoint requires a separately checked infeasibility witness. The intervals in Table 7 are not such witnesses.

For a zero-effect target, the ratio is undefined and absolute errors remain the appropriate report. For unequal numerical queries, the perturbation term is evaluated on the actual query discrepancies. Multihead compensation, earlier-layer prefix effects, shifted positions, and downstream token predictions remain outside this fixed-head comparison.

Table 10: Rank-four resource sweep. Errors are normalized by the fixed target effect. The at-most values include all smaller counts and the zero-slot option; equal count-1 and count-4 optima give the displayed minima, and no better intermediate count is assumed.

Head	Target	Scale	m	K	V	Exact-count	Learned	At most
0	Value	0.5	1	4.20	3.05	0.624	0.653	0.624
0	Value	0.5	4	4.20	3.05	0.624	0.658	0.624
0	Value	0.5	16	4.20	3.05	0.670	0.728	0.624
0	Value	1.0	1	8.40	6.10	0.407	0.436	0.407
0	Value	1.0	4	8.40	6.10	0.407	0.441	0.407
0	Value	1.0	16	8.40	6.10	0.407	0.465	0.407
0	Value	2.0	1	16.80	12.20	0.351	0.380	0.351
0	Value	2.0	4	16.80	12.20	0.351	0.385	0.351
0	Value	2.0	16	16.80	12.20	0.351	0.409	0.351
4	Query	0.5	1	5.10	3.65	0.511	0.540	0.511
4	Query	0.5	4	5.10	3.65	0.511	0.545	0.511
4	Query	0.5	16	5.10	3.65	0.557	0.615	0.511
4	Query	1.0	1	10.20	7.30	0.319	0.348	0.319
4	Query	1.0	4	10.20	7.30	0.319	0.353	0.319
4	Query	1.0	16	10.20	7.30	0.319	0.377	0.319
4	Query	2.0	1	20.40	14.60	0.273	0.302	0.273
4	Query	2.0	4	20.40	14.60	0.273	0.307	0.273
4	Query	2.0	16	20.40	14.60	0.273	0.331	0.273