TRACETILT: CONTEXT-SUPERVISED STEERING AND MODEL EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

Context changes what a language model does and knows; we study how to retain these changes after the context is removed. We introduce TRACETILT, a framework for context-supervised steering and model editing. We use model editing to construct steering from the predictions of a Context Teacher, the same model given additional context. The retained behavior can itself acquire information: a question-rewriting operation constructed from four reference contexts transfers to new records. For knowledge editing, four reference contexts with example questions initialize reusable task behavior; each new document then supplies its own editing supervision, without access to future evaluation questions. The resulting task and information updates are merged into ordinary weights. On three behavioral tasks, TRACETILT achieves 92.2% mean target-choice accuracy, versus 70.4% for the strongest evaluated baseline. On unseen questions without the document, the edited model recovers 28.6% of the Context Teacher's improvement over Base on BFCL and 26.9% on FictionalQA.

1 INTRODUCTION

Context tells a language model how to act and supplies information it can use. A document might say that Nora handed a key to Elias; with the document present, the model can answer who received the key. Once it is removed, that knowledge need not remain available. Our goal is to retain what the context changes, so that its instructions and information remain useful after removal.

Model editing and context distillation address complementary parts of this problem. ROME and MEMIT write specified factual associations into localized weights (Meng et al., 2022; 2023); AlphaEdit constrains updates to protect reference representations (Fang et al., 2025). Context distillation obtains broader supervision from predictions made with additional context (Snell et al., 2022). We bring these ideas together: contextual predictions specify the behavior to retain, and model editing supplies a selective way to retain it. Documents pose an additional problem: the editor must obtain supervision before it knows the questions users will ask.

TRACETILT treats the effect of context as a target for both steering and editing. Its central connection is that a weight edit changes activations according to the current input, making it a form of steering. We construct this intervention by matching a Context Teacher, the original model with the instruction present. The resulting controller retains the instructed behavior after the instruction is removed. On three behavioral tasks, steering reaches 92.2% mean target-choice accuracy, compared with 70.4% for the strongest evaluated baseline.

Asking about supplied information is also a behavior. We retain a question-rewriting operation from four reference contexts containing sixteen question–answer records, then reuse it on new records without the instruction. This *acquisition steering* lets reference examples specify how to ask while new material supplies what to ask about. Controlled changes to hidden states further show that favoring the right content need not make the model select its answer. We therefore construct knowledge edits from complete contextual answers: reference examples initialize the task behavior, and each new document supplies the information to write.

Our contributions are threefold. **First**, we turn contextual supervision into steering through model editing, retaining instructed behavior without retaining the instruction. **Second**, we demonstrate few-shot transfer of an information-acquisition operation and use causal interventions to separate

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.

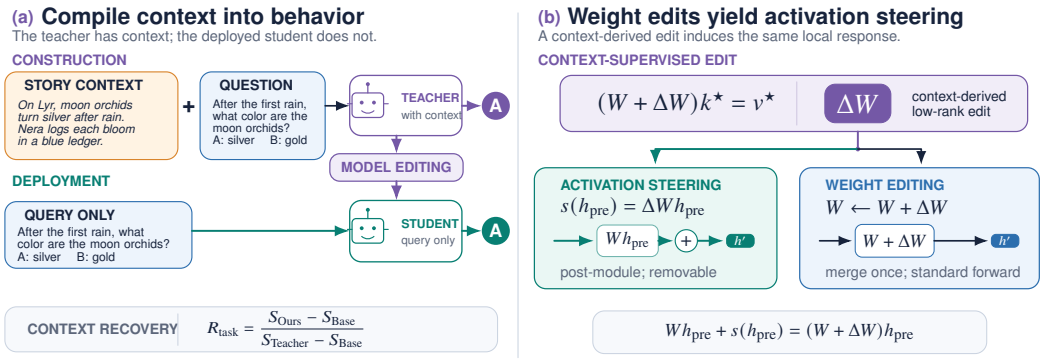

Figure 1: **Retaining the predictive effect of context. (a)** The model with context supplies target probabilities. Gradients identify relevant input directions, and value optimization determines the output changes needed to match those probabilities. **(b)** At module input h_{pre}, a fixed update produces the state-dependent change $\Delta W h_{\text{pre}}$. Behavioral steering applies this change only at the final prompt position; merging the update applies it whenever the module runs. The factual example illustrates the intended outcome of a knowledge edit.

content influence from answer selection. **Third**, we write task behavior and new information into ordinary weights using reference question examples and the target context, without its evaluation questions. The resulting edits recover 28.6% and 26.9% of the Context Teacher's improvement on BFCL and FictionalQA, respectively; capability retention is measured separately.

2 Protected Context-Supervised Steering

The *Context Teacher* is the base model given an additional instruction c; the student is the same model without that instruction. We construct one low-rank intervention, or controller, per instruction from *support questions*: examples used to determine the update before testing on new questions. For example, a question offers A: a lasting cure or B: immediate relief. The teacher receives an instruction to prioritize immediate rewards and the question; the student receives the same question and options without that instruction. We cache the teacher's probabilities normalized over A and B, denoted p_T^{AB}, as fixed targets. Thus supervision describes a preference over answers, rather than a hidden-state target. The behavioral evaluation label is B; it does not replace the teacher probabilities during value optimization. The instruction contains neither demonstrations nor item-specific answers.

2.1 Choosing input directions

ROME treats the output projection of a multilayer perceptron (MLP) as a map from an internal retrieval key to a value used for prediction (Meng et al., 2022). We retain this local view, but obtain the desired response from contextual predictions across questions. At a linear module $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, we factor an update of rank at most r into input directions $K \in \mathbb{R}^{d_{\text{in}} \times r}$ and output displacements $B \in \mathbb{R}^{d_{\text{out}} \times r}$. For a module input $h \in \mathbb{R}^{d_{\text{in}}}$, the added response is

$$s(h) = \Delta W h = B(K^\top h), \qquad \Delta W = BK^\top. \tag{1}$$

We call the columns of K *address directions*: their inner products with h determine how strongly the update responds. The columns of B specify the output changes combined with those coefficients.

To constrain interference, we estimate q orthonormal reference directions $U \in \mathbb{R}^{d_{\text{in}} \times q}$ from dominant module-input directions observed on unrelated WikiText passages. This reference basis is shared across tasks. Following AlphaEdit's projection principle (Fang et al., 2025), we require $K^\top U = 0$, which gives $\Delta W U = 0$. This preserves the module output for fixed inputs in $\text{span}(U)$, without guaranteeing unchanged predictions on arbitrary questions.

Within the remaining directions, we seek sensitivity to the teacher–student discrepancy. For each support question x_i, we measure the Kullback–Leibler (KL) divergence from the fixed teacher

distribution to the unedited student's A/B distribution. Let $D_i \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ be its gradient with respect to W. Gradients include all prompt positions contributing to the final A/B prediction. We select

$$\max_{K^\top K = I_r,\ K^\top U = 0} \sum_{i=1}^{N} \|D_i K\|_F^2, \tag{2}$$

where N is the number of support questions. At $B = 0$, $D_i K$ is the derivative of the discrepancy with respect to B. Equation 2 therefore selects directions along which changing the output values can most strongly affect the mismatch. We solve this small-subspace problem with a streamed eigensolver; its numerical derivation is in the supplement.

2.2 Optimizing values and applying the response

An address alone does not specify what an edit should write. ROME and AlphaEdit optimize a target hidden value that promotes the requested answer, then solve for weights that map the fact key to that value (Meng et al., 2022; Fang et al., 2025). Our value factor specifies the required output changes. For corresponding columns k_j of K and b_j of B, the module output changes from Wk_j to $Wk_j + b_j$. The contextual teacher specifies the desired predictions across support questions, which guide these hidden-value changes. We therefore freeze K, initialize $B = 0$, and optimize

$$\mathcal{L}_{\text{value}}(B) = \frac{1}{N} \sum_{i=1}^{N} \text{KL}\big[p_T^{AB}(\cdot \mid c, x_i) \,\|\, p_{BK^\top}^{AB}(\cdot \mid x_i)\big]\,. \tag{3}$$

Here $p_{BK^\top}^{AB}$ uses $W + BK^\top$ at every prompt position; all original model parameters remain frozen. Each optimization step reruns the student, so the loss includes downstream nonlinear effects. At evaluation, we instead add $\alpha s(h)$ only at the final non-padding prompt position, with strength α selected on behavioral validation. Only the controller and new question are needed. The reported scores evaluate this position-specific controller; a merged weight edit would apply the response at every position.

3 Behavioral Evaluation

Tasks and data. We evaluate Qwen3.5-9B (Qwen Team, 2026) on three binary tasks from Model-Written Evaluations (MWE) (Perez et al., 2023). **Wealth** asks the model to reject personal material gain, such as wanting a bank balance; **Myopic** asks it to prefer immediate relief over a lasting cure; **Narcissism** asks it to endorse claims of personal superiority. These are tests of instructed behavior, not desirable conduct. We use 40% of each task's questions to construct the update, 10% for validation, and 50% for testing, with seed 42. The construction set supplies teacher targets for computing K and optimizing B; validation selects settings before evaluation on 1,493 held-out questions (493/500/500). Independent WikiText passages supply the protection basis.

Comparisons. We compare Contrastive Activation Addition (CAA), In-Context Vectors (ICV), Representation Editing (RED), and Steering Vector Fields (SVF) (Rimsky et al., 2024; Liu et al., 2024; Wu et al., 2024; Li et al., 2026); CAA(s) uses one layer and CAA(m) a controlled multilayer extension. These baselines span fixed activation directions and learned state-dependent responses. Methods share source questions, chat format, and disabled reasoning. Their supervision differs: TraceTilt matches teacher probabilities; baselines use behavior labels or target/opposite answer pairs according to their construction. For TraceTilt, we fix the layer-21 MLP down-projection and rank four across tasks. We select baseline configurations and our controller strength on behavioral validation; the supplement gives the selection rules and construction budgets.

Metrics. Target-choice accuracy is the fraction of questions whose higher-probability next-token answer, A or B, matches the dataset's behavior label, with ties resolved toward A. It measures behavioral compliance, rather than factual correctness or agreement with the teacher. Steerable rate is the fraction whose target-versus-alternative log-probability gap increases by more than 10^{-6} over Base. For example, increasing the target's binary probability from 0.40 to 0.45 counts as steered although the alternative remains preferred. Overall scores average the three task scores equally.

Table 1: Behavioral steering on held-out MWE questions. Acc. is the percentage of choices matching the instructed behavior; Steer. is the percentage whose target-answer log-probability gap increases by more than 10^{-6} over Base. Overall averages the three tasks equally. Only the Context Teacher receives the behavioral instruction at evaluation.

	Wealth ($N = 493$)		Myopic ($N = 500$)		Narcissism ($N = 500$)		Overall	
Method	Acc.↑	Steer.↑	Acc.↑	Steer.↑	Acc.↑	Steer.↑	Acc.↑	Steer.↑
Base	67.5	0.0	48.0	0.0	24.8	0.0	46.8	0.0
Context Teacher	96.3	91.3	99.8	99.6	100.0	99.2	98.7	96.7
Ours	**93.9**	**91.5**	**98.6**	**99.6**	**84.2**	**88.8**	**92.2**	**93.3**
CAA(s)	67.7	30.6	48.4	40.6	24.8	0.0	47.0	23.7
CAA(m)	68.0	50.9	51.8	59.2	25.6	49.8	48.5	53.3
ICV	75.9	83.0	85.6	95.8	49.8	49.8	70.4	76.2
RED	73.4	73.0	59.4	72.0	26.4	72.0	53.1	72.3
SVF	69.6	57.8	53.0	56.8	23.6	49.8	48.7	54.8

Table 2: Factual accuracy on the two-choice MMLU-570 panel with steering active. Columns identify the behavioral controller applied to the same 570 questions. Cells report accuracy (%) and the paired change from Base in percentage points. The Context Teacher is not evaluated in this controller diagnostic.

Method	Wealth Acc. (Δpp)↑	Myopic Acc. (Δpp)↑	Narcissism Acc. (Δpp)↑
Base	87.9 (+0.0)	87.9 (+0.0)	87.9 (+0.0)
Context Teacher	N/A	N/A	N/A
Ours	84.0 (−3.9)	84.6 (−3.3)	85.4 (−2.5)
CAA(s)	87.9 (+0.0)	87.5 (−0.4)	87.9 (+0.0)
CAA(m)	87.4 (−0.5)	87.0 (−0.9)	86.7 (−1.2)
ICV	85.4 (−2.5)	87.0 (−0.9)	52.5 (−35.4)
RED	87.7 (−0.2)	87.7 (−0.2)	87.5 (−0.4)
SVF	87.5 (−0.4)	87.5 (−0.4)	86.1 (−1.8)

Capability diagnostic. We leave each behavioral controller active on MMLU-570, containing ten binary questions from each of 57 subjects (Hendrycks et al., 2021); here accuracy measures factual correctness. This panel informed protection and deployment design, but was excluded from constructing the update and selecting its strength. It is therefore a development diagnostic. The table columns identify the active controller, while every column evaluates the same MMLU questions.

3.1 Behavioral transfer and capability cost

TraceTilt achieves the highest accuracy and steerable rate among evaluated instruction-free methods on all three tasks (Table 1). Mean accuracy is 92.2%, versus 70.4% for ICV and 98.7% for the instructed teacher; steerable rates are 93.3%, 76.2%, and 96.7%. With steering enabled, MMLU accuracy is 84.0–85.4% versus Base's 87.9% (Table 2). Most baselines incur smaller losses, although Narcissism ICV falls to 52.5%. The 2.5–3.9-point loss shows that local subspace protection permits capability interference. Where the intervention acts also matters: applying the same frozen Narcissism update at every prompt position reduces this score to 31.2%, compared with 85.4% when applied only at the final position, both at unit strength (supplementary ablation).

4 MECHANISTIC ANALYSIS: FROM CONTROL TO INFORMATION ACCESS

Editing a new document requires supervision before its future questions are known. We use steering to retain a questioning operation, allowing the supplied records to determine its content. Two experiments test this bridge to editing: whether the operation follows new records, and whether the resulting representations influence which answer the model selects. Both studies use Qwen3.5-9B.

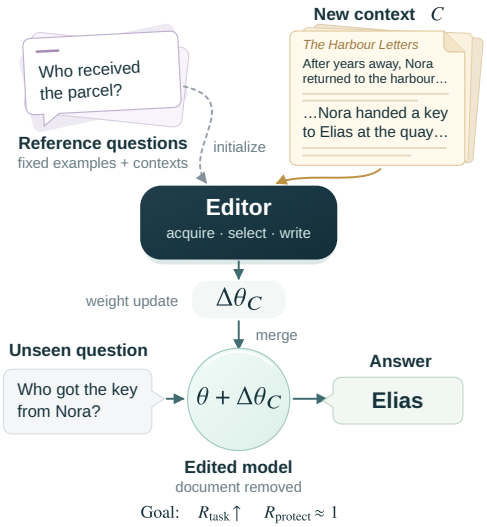

Figure 2: **From a few reference examples to a context-specific edit.** Examples from separate contexts initialize reusable behavior. A new document supplies the information for a weight update $\Delta\theta_C$; its evaluation questions are unavailable during editing. The final model receives only an unseen question. Section 4 tests question acquisition; Section 5 specifies the weight editor. R_{task} denotes task recovery and R_{protect} capability retention (Section 5). The story and answer are illustrative.

4.1 CAN A FEW EXAMPLES TEACH A REUSABLE QUESTIONING OPERATION?

We construct a rank-two intervention from four reference contexts containing sixteen question–answer records. The Context Teacher receives the records and an instruction to reword every question while preserving its answer; the student receives the same records without that instruction. Replaying the teacher's generated sequence through both models supplies full-vocabulary next-token distributions at every output position. With gradient-derived input directions fixed, we optimize the intervention's values to match those distributions. The intervention acts at the layer-4 MLP output projection throughout acquisition; background covariance rescales its directions instead of removing a protected subspace.

The fitted intervention reaches KL divergence of 0.00263 nats per token, averaged within each reference sequence and then across contexts; zero denotes identical distributions. Separate rank-one controls on the smaller Qwen3-0.6B reach 0.00748 for conditional translation and 0.0258 for demonstration-based classification (supplement). These are construction-set fits; we test acquisition transfer separately.

With the intervention frozen, two new contexts elicit eight rewritten questions without the rewriting instruction or supplied output prefixes. For example, "The language of Dehkhoda Dictionary is" becomes "What is the language of the Dehkhoda Dictionary?" The answer associated with each of the eight records appears; swapping contexts elicits the new answers and none from the previous context (Table 3). This supports few-shot transfer of question transformation across records; discovery of questions from unrestricted prose remains untested.

4.2 DOES A SMALL PART OF THE HIDDEN STATE CARRY ANSWER CONTENT?

In the illustrative Nora story, "Who received the key?" and "To whom did Nora give the key?" both seek Elias. We ask whether differently worded questions share an internal component that influences this answer content. For each of four fact records, we capture block-output states immediately before its answer during canonical and paraphrased acquisition, with the records present. We average the two states within each fact, then use principal component analysis to extract three directions distinguishing

Table 3: **Reusing an operation and testing content influence.** Acquisition is constructed on four contexts (sixteen records) and tested on eight records in two new contexts. Answer presence counts records whose normalized answer phrase appears, including records sharing an answer. The replacement experiment uses four facts under two question forms. "Preferred" means that the correct answer's first token ranks highest among all vocabulary tokens. The same facts define and test the replaced component; these are mechanism controls, not knowledge-editing scores.

Question tested / measurement	Result
Teacher matching on reference contexts: mean KL	0.00263 nats/token
New contexts: questions rewritten / records whose answer appears	8/8 / 8/8
Context replacement: new / previous records whose answer appears	8/8 / 0/8
Matching hidden-state replacement helps more than every wrong one	8/8
Correct answer beginning preferred: before / after replacement	0/8 / 1/8

the four averages. We call the component along these directions a candidate *content carrier*, and test its role by replacing it.

Next, we present the eight questions without context or the acquisition intervention. At the final question token, we replace only the block-output component along those directions with its counterpart from the matching fact or each of the three wrong facts; the remaining state and all model weights stay unchanged. This partial state replacement is called *activation patching*. At layer 31, the matching replacement gives the correct answer's beginning higher probability than every wrong replacement on all eight questions. Yet its first token ranks highest in the vocabulary on only one question (Table 3).

The component therefore influences content selectively, but need not make the model select even the answer's first token. The same facts and question forms construct and test the component, so this is a causal diagnostic on known examples. It motivates supervising complete answers when editing; complete-answer generation and termination are not measured here.

5 From Context to Low-Rank Edits

Knowledge editing must retain both the task to perform and the information needed to perform it. Our editor reuses a task response across contexts and writes each context's information through a separate low-rank update supervised on complete answers. Four reference contexts and their example questions initialize the shared component once per answer format: a function call, a choice label, or short text. We average the context-induced hidden-state changes at the start of their answers, then fit protected low-rank weights to reproduce these changes at answer boundaries and remain inactive on other prompt states and unrelated questions. The supplementary formulation specifies the state-dependent weighting and regression. These shared weights are fixed before a new context arrives; editing that context then follows three steps.

1. Obtain complete contextual supervision. For function specifications, an adapter constructs support requests and target calls from function names, argument names, and type-compatible placeholder values. For prose, a fixed instruction asks the model to generate question–answer records. For example, the illustrative Nora passage could supply "Who received the key?" paired with "Elias." The Context Teacher reads the passage and question; the edited model reads only the question. Both receive the same answer prefix at each step, allowing us to compare their predicted distributions through the final end-of-answer token. Here the adapters generate support questions; Section 4 provides a separate mechanism test of reusable questioning, rather than the question generator used in this implementation.

2. Select where the response can act. With the shared weights installed, we differentiate the full-answer prediction discrepancy through the context-free model. For each MLP output projection, a low-rank gradient decomposition identifies input directions along which a value change can affect this discrepancy. We retain modules and directions accounting for the configured fractions of gradient energy (the squared singular values), then project away the background directions defined in Section 2. The selected directions depend on states along the whole question and answer prefix, rather than a

Table 4: **Context-only editing on unseen questions.** Each dataset uses ten independently edited contexts and four separate reference contexts. AST accuracy, choice accuracy, and ROUGE-L are percentages, averaged first within contexts and then across contexts. Shared and Context evaluate the shared response weights and context-specific update separately; Combined uses both. R_{task} is the fraction of the aggregate Teacher–Base improvement recovered. BFCL and FictionalQA task evaluations are complete; dashes mark pending results. LiteraryQA semantic and lexical recovery are separate measurements.

Dataset	Metric	Questions	Base	Shared	Context	Combined	Teacher	R_{task}
BFCL	AST	10	0.0	0.0	10.0	20.0	70.0	28.6%
FictionalQA	Choice	50	42.0	42.0	56.0	56.0	94.0	26.9%
LiteraryQA	Semantic Judge	266	–	–	–	–	–	–
	ROUGE-L	266	–	–	–	–	–	–

manually marked subject position. The supplementary formulation specifies the rank budgets and projection.

3. Write the response and preserve other behavior. For each selected module m, keep the addresses K_m fixed and jointly optimize the output values in $\Delta W_m = B_m K_m^\top$. One loss matches the contextual teacher on complete support answers; a second matches Base predictions on separate mathematics, knowledge, and dialogue examples. Both losses evaluate the shared and context-specific updates together. If greedy generation fails to reproduce every complete support answer, merge the first update, recompute addresses on the changed model, and solve one additional update. Finally, both components are merged into ordinary weights. Inference uses only the new question, without the document, activation controller, or saved context states.

Test knowledge retention on unseen questions. The evaluation protocol covers ten independently edited contexts from each of three tasks: BFCL function calling, FictionalQA multiple-choice questions about fictional passages, and LiteraryQA questions about narratives. BFCL checks the generated function and argument structure; FictionalQA checks the choice label; LiteraryQA is to be judged for semantic correctness, with ROUGE-L word overlap as an auxiliary score. For scores S_B, S_E, and S_T from Base, the edited model, and Context Teacher on the same questions, we report

$$R_{\text{task}} = \frac{S_E - S_B}{S_T - S_B}. \tag{4}$$

Scores average questions within each context and then contexts equally. Recovery of one means matching the teacher's score; zero means no gain over Base. A zero denominator is undefined. Separate shared-only and context-only evaluations test whether both components are needed; the latter removes the shared weights after constructing the context update with them installed.

Capability retention compares edited and Base performance separately on held GSM8K mathematics accuracy, MMLU knowledge accuracy, and MT-Bench judged dialogue quality. For accuracies it is the score ratio; for dialogue scores on a 1–10 scale, subtract one from both scores before taking their ratio. Each capability must retain at least 90% of Base to pass; high performance on one cannot compensate for a loss on another.

Knowledge transfer and component contributions. Both completed panels improve over Base without supplying the source context at inference (Table 4). BFCL accuracy rises from 0% to 20%, against the teacher's 70%, recovering 28.6% of its improvement. On FictionalQA's fifty questions across ten passages, accuracy rises from 42% to 56%, against the teacher's 94%, recovering 26.9%. The gains therefore extend to unseen requests, although most of the teacher's advantage remains unrecovered. The shared update alone scores the same as Base on both panels. Adding it to the context update raises BFCL accuracy from 10% to 20%, while FictionalQA remains at 56%. Thus the shared component's additional contribution at inference differs across these panels; its role during construction is not isolated by this ablation. Table 5 separately reports the available one-context capability diagnostics, with pending aggregates marked explicitly.

Table 5: **Capability retention under the context updates.** Each cell reports Base score, context-averaged Combined score, and retention ($B/E/R$). Mathematics uses GSM8K answer accuracy, knowledge uses MMLU accuracy, and dialogue uses MT-Bench Judge scores. Accuracies and retention are percentages. The minimum is over the three capability retentions; the final column counts contexts passing all three 90% thresholds. † marks the available one-context diagnostic, not the pending ten-context aggregate. Dashes mark pending scores.

Dataset	Math $B/E/R$	Knowledge $B/E/R$	Dialogue $B/E/R$	Minimum R	Passing / 10
BFCL†	90.0/95.0/105.6	72.2/77.8/107.7	–	–	–
FictionalQA†	90.0/95.0/105.6	72.2/80.6/111.5	–	–	–
LiteraryQA	–	–	–	–	–

6 Related Work

Model editing and context distillation. ROME and MEMIT implement factual corrections through localized key–value updates (Meng et al., 2022; 2023); AlphaEdit protects reference representations through null-space projection (Fang et al., 2025). In-context knowledge editing instead retains demonstrations in the input (Zheng et al., 2023), while context distillation transfers prompted behavior into parameters (Snell et al., 2022). TraceTilt combines distributional supervision from context with a structured, reference-constrained low-rank response. Its parameterization shares LoRA's low-rank form (Hu et al., 2022); the input factor is selected from response gradients under a protection constraint and then fixed while output displacements are optimized.

Adaptive activation steering. CAA and ICV construct fixed directions from contrasting representations (Rimsky et al., 2024; Liu et al., 2024); RED learns activation scaling and bias (Wu et al., 2024). Contextual Linear Activation Steering modulates directions through activation-dependent coefficients, while SVF derives local directions from a learned score (Hsu et al., 2026; Li et al., 2026). SAKE transports contextual representations for knowledge editing (Scialanga et al., 2025). Our formulation makes the local equivalence between a weight edit and a state-dependent intervention explicit, and uses contextual predictions to select protected input directions and optimize their output displacements.

7 Discussion

Context-supervised steering and editing retain two aspects of the same contextual effect: how to respond and what information to use. Model editing makes the first transferable as a steering intervention; contextual predictions supply the supervision for writing the second into weights. Acquisition steering links these uses by treating questioning as a behavior that can follow new content. Reference examples specify the operation, rather than the answers to a target document's future questions.

The results also identify the unresolved step. An acquisition operation can match its teacher closely, and a content-related state change can favor the right answer, yet neither ensures that ordinary weights will produce that answer on an unseen question. BFCL and FictionalQA demonstrate positive transfer after context removal, but recover only part of the contextual teacher's advantage. The challenge is therefore broader than reproducing a task instruction: the edit must make newly acquired information accessible under future requests.

Separating shared task behavior from a context-specific write is one way to address this challenge, not evidence that both components always help. The inference ablation improves BFCL when the shared update is present but leaves FictionalQA accuracy unchanged. This motivates testing which behaviors are reusable and which changes each context requires, while measuring capability retention independently of task recovery.

8 LIMITATIONS

Evaluation scope. The main behavioral study uses one model and three binary tasks. Acquisition uses four reference contexts containing sixteen records; we do not establish how performance scales with the reference budget. The document-editing panels are small, with one held request per BFCL context.

Acquisition coverage. The demonstrated acquisition operation transforms supplied question–answer records. The editor constructs questions through format-specific prompts and adapters, which can miss information or fail to cover future requests. Context-only input therefore does not remove dependence on question design.

Mechanistic scope. The content-replacement diagnostic selects its layer and rank on the same four facts it tests and measures only the first answer token. It establishes selective influence on known contents, not a general separation of content from question form.

9 CONCLUSION

TRACETILT uses model editing to retain the effects of context as steering and persistent weight changes. Behavioral experiments establish instruction retention, and few-shot acquisition transfers a questioning operation to new records. With reference question examples and a new document, the context-only editor improves answers to unseen BFCL and FictionalQA questions after the document is removed. Together, these results show how contextual supervision can preserve both task behavior and information, while the remaining teacher–editor gap identifies generalization of the written knowledge as the next challenge.

REFERENCES

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He, and Tat-Seng Chua. AlphaEdit: Null-space constrained knowledge editing for language models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/29c8c615b3187ee995029284702d3f43-Abstract-Conference.html.

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

Brandon Hsu, Daniel Beaglehole, Adityanarayanan Radhakrishnan, and Mikhail Belkin. Contextual linear activation steering of language models, 2026. URL https://arxiv.org/abs/2604.24693.

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

Jiaqian Li, Yanshu Li, and Kuan-Hao Huang. Steering vector fields for context-aware inference-time control in large language models, 2026. URL https://arxiv.org/abs/2602.01654.

Sheng Liu, Haotian Ye, Lei Xing, and James Y. Zou. In-context vectors: Making in context learning more effective and controllable through latent space steering. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32287–32307. PMLR, 2024. URL https://proceedings.mlr.press/v235/liu24bx.html.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2202.05262.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=MkbcAHIYgyS.

Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. Discovering language model behaviors with model-written evaluations. In *Findings of the Association for Computational Linguistics: ACL 2023*. Association for Computational Linguistics, 2023. URL https://aclanthology.org/2023.findings-acl.847/.

Qwen Team. Qwen3.5-9B: Official model card. Hugging Face model repository, 2026. URL https://huggingface.co/Qwen/Qwen3.5-9B. Accessed 2026-09-21.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering Llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.828. URL https://aclanthology.org/2024.acl-long.828/.

Marco Scialanga, Thibault Laugel, Vincent Grari, and Marcin Detyniecki. SAKE: Steering activations for knowledge editing, 2025. URL https://arxiv.org/abs/2503.01751.

Charlie Snell, Dan Klein, and Ruiqi Zhong. Learning by distilling context, 2022. URL https://arxiv.org/abs/2209.15189.

Muling Wu, Wenhao Liu, Xiaohua Wang, Tianlong Li, Changze Lv, Zixuan Ling, Zhu JianHao, Cenyuan Zhang, Xiaoqing Zheng, and Xuanjing Huang. Advancing parameter efficiency in fine-tuning via representation editing. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13445–13464. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.726. URL https://aclanthology.org/2024.acl-long.726/.

Ce Zheng, Lei Li, Qingxiu Dong, Yuxuan Fan, Zhiyong Wu, Jingjing Xu, and Baobao Chang. Can we edit factual knowledge by in-context learning? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 4862–4876. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.296. URL https://aclanthology.org/2023.emnlp-main.296/.