# CHARACTERIZING THE GAP BETWEEN KNOWING AND ANSWERING IN SPEECH LANGUAGE MODELS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

A large speech language model can learn a fact from text and answer a written question correctly, but still fail to answer the same question through speech. This discrepancy does not establish whether the learned information is absent from speech processing, is present but not affecting the response, or contributes to a response that still fails to recover the answer. We investigate these possibilities by teaching models new facts through text-only fine-tuning and examining how this learning changes their hidden states when processing speech. By changing the learned answers and selectively removing these changes, we test whether they reflect answer content and influence generation. Our results show that information about the learned answer can remain present and affect the model's response even when speech recall fails. On the speech failures we examine, removing the changes associated with the queried fact harms responses more than removing equally strong changes associated with other facts. We observe this content-sensitive effect across two model families and under changes in question wording and synthesized voice. However, in a controlled ablation study, increasing the measured changes beyond their original strength provides little additional improvement in average answer quality. These findings show that a model can use knowledge learned from text and still answer a spoken question incorrectly. Evidence that this knowledge contributes to a response does not establish that simply strengthening its contribution will produce the correct answer.

## 1 INTRODUCTION

Recent advances in large speech language models (LSLMs) have enabled users to interact with AI systems directly through speech (Zhang et al., 2023; Défossez et al., 2024; Chu et al., 2024; Xu et al., 2025), extending human-computer interaction from typed text to more natural spoken dialogue. Compared to text input, speech input provides additional information, such as emotion, prosodic cues, and background sounds (Défossez et al., 2024; Ao et al., 2024), which can help models understand the speaker's intended meaning and determine how to respond appropriately (Ao et al., 2024; Wang et al., 2025). To answer spoken questions effectively, LSLMs need to understand not only what users say but also the additional information conveyed through speech (Ao et al., 2024; Wang et al., 2025), and use relevant knowledge to provide accurate answers (Tang et al., 2024; Défossez et al., 2024). However, many LSLMs are built on pretrained text language models (Tang et al., 2024; Défossez et al., 2024; Cuervo et al., 2026), which raises the question of whether text-learned knowledge can be retrieved and used to generate responses to spoken input. Empirical studies report gaps between speech and text performance (Xiang et al., 2025; Cuervo et al., 2026; Wang et al., 2026b) and differences in factual recall mechanisms across the two modalities (Modica et al., 2026). These observations raise a critical question: when a model fails to recall through speech a fact it learned from text, what does this failure reveal about the learned knowledge?

When a model responds well to text input but produces a poorer response to the same content provided through speech (Xiang et al., 2025; Wang et al., 2026b; Hsu et al., 2026), it is unclear from the response alone why changing the input modality affects its quality. One possibility is that the knowledge learned from text does not contribute to generating the answer from speech, a concern motivated by studies of incomplete cross-modal transfer and alignment (Modica et al., 2026; Cuervo et al., 2026; Wang et al., 2026b). Another is that this knowledge does influence the model's response, but its contribution is not enough to produce the correct answer. Related evidence comes from text



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.

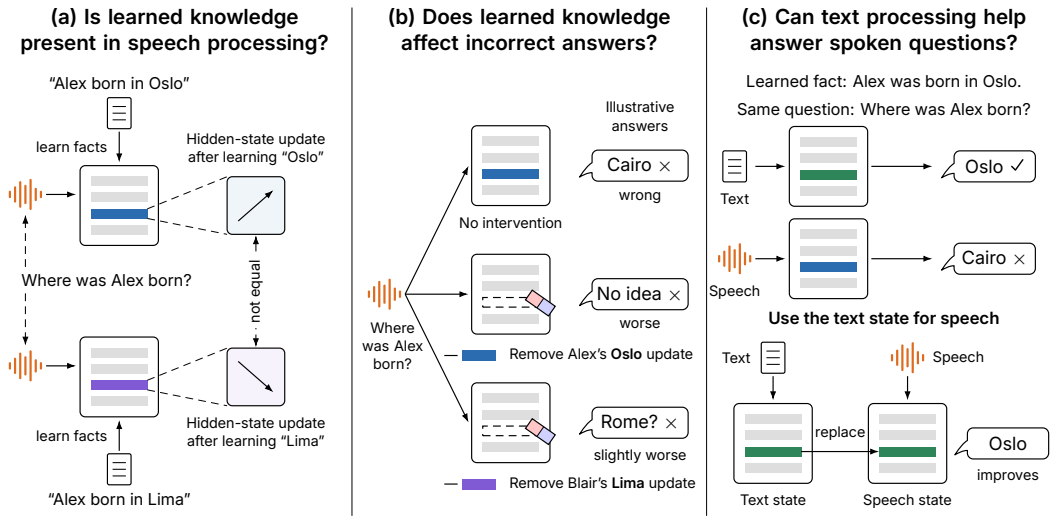



Figure 1: **Three questions about how speech language models use knowledge learned from text.** (a) Does the answer learned from text affect the model's hidden states when it processes the same spoken question? (b) Do these changes contribute to the response even when the answer is incorrect? (c) Can replacing a speech hidden state with the corresponding text hidden state improve the answer?

models whose internal truthfulness signals diverge from their generated answers (Orgad et al., 2025; Li et al., 2023), and from speech models whose intermediate representation alignment does not ensure correct late-layer decisions (Hsu et al., 2026). The first possibility directs attention to how this knowledge can contribute to generating answers from speech, while the second raises the question of why an existing contribution remains insufficient for successful recall. Distinguishing these possibilities requires evidence beyond output accuracy, including interventions that test whether internal information affects model behavior (Elazar et al., 2021; Geva et al., 2023). Decodable information need not be used by the model (Belinkov, 2022; Elazar et al., 2021), and improving an answer by supplying an activation from another computation (Ghandeharioun et al., 2024; Zhang & Nanda, 2024) does not by itself establish what information the original speech computation used. More broadly, effective intervention sites need not coincide with those identified by causal localization (Hase et al., 2023). We therefore ask whether knowledge learned from text still contributes to the model's response when it fails to answer a spoken question correctly (Figure 1).

To understand what failed recall reveals about learned knowledge, we examine three questions. First, does information about the specific answer learned from text remain in the model's internal states when it processes a spoken question? Second, does this information contribute to the model's response even when the answer is incorrect? Third, is strengthening the changes associated with this information sufficient to produce the correct answer? To investigate these questions, we fine-tune a pretrained LSLM on text question–answer pairs and compare its hidden states before and after fine-tuning using the same speech input. We call the difference between these hidden states the *speech-update vector*, which represents how text-based learning changes the model's internal states when processing speech. By comparing vectors associated with different learned answers and testing their effects on generation, we examine whether these changes reflect the learned answer and contribute to the response. We then test whether restoring or strengthening the vector can improve failed answers.

Our results show that knowledge learned from text can still contribute to a model's response even when it fails to answer a spoken question correctly. When we change the answers used for fine-tuning while keeping the questions the same, the effects of the speech-update vectors change with the learned answers. Removing the vector for the same question also reduces the model's preference for the correct answer and harms its response more than removing a vector from another question. We observe similar effects under the tested changes in model family, question wording, and synthesized voice. However, increasing the strength of the vector does not necessarily improve the answer. On the tested failures, restoring the removed vector improves average answer quality, but

increasing its strength beyond the original level provides little additional improvement. Replacing the speech hidden state with one obtained from the corresponding text input can still improve these responses. Together, these findings show that a poor response to speech does not necessarily mean that the model is not using knowledge learned from text. This knowledge may already contribute to the response, while simply strengthening the measured vector is not enough to produce a correct answer.

## 2 Related Work

**Cross-modal memorization and speech–text alignment.** Shared representation spaces can retain modality-specific structure (Liang et al., 2022; Jiang et al., 2023; Yaras et al., 2025). Controlled single-modality updates reveal incomplete and directional cross-modal memorization in vision-language models (Wen et al., 2025). For speech-language models, targeted angle and length interventions connect representation geometry to answer quality (Xiang et al., 2025), while distillation addresses text-capability forgetting, output mismatch, and reasoning differences (Cuervo et al., 2026; Wang et al., 2026b;a; Hu et al., 2026). These studies examine how knowledge transfers across modalities and how speech responses can be improved. However, improvements in response quality do not by themselves reveal whether knowledge learned from text already contributes to the responses that remain incorrect. We investigate this question by examining the role of learned information during failed speech recall.

**Task representations and model differences.** Comparing representations across modalities or between models before and after fine-tuning can reveal how information is shared and what changes during learning (Luo et al., 2025; Lindsey et al., 2024; Minder et al., 2025). Task vectors and function vectors can encode task information from demonstrations and guide responses to new inputs (Hendel et al., 2023; Todd et al., 2024). Multimodal task and function vectors also support learning from image–text demonstrations and controlling visual task behavior (Huang et al., 2024; Fu et al., 2026), while cross-modal task vectors extend this idea to transferring task information between image and text inputs (Luo et al., 2025). Model diffing uses activation comparisons to identify shared and changed features (Lindsey et al., 2024; Jiralerspong & Bricken, 2026; Kassem et al., 2026), and activation differences can reveal the content and format of a model's fine-tuning data (Minder et al., 2025). Related methods identify internal features that causally affect fine-tuned behavior (Kassem et al., 2026) or recover learned facts by comparing the output distributions of the original and fine-tuned models (Brzozowski et al., 2026). These approaches provide ways to identify what a model has learned and how learning changes its behavior. We build on this idea to examine what those changes do when knowledge is queried through speech and the model answers incorrectly. Our question is whether the measured changes reflect the learned answer and whether they still contribute to the failed response.

**Factual mechanisms and internal intervention.** Studies of model internals examine what information a model represents, how that information affects its output, and whether changing it can improve or alter the answer. Probing methods test whether information can be recovered from hidden states (Belinkov, 2022; Hewitt & Liang, 2019; Orgad et al., 2025), but a learned probe may solve part of the task itself (Hewitt & Liang, 2019), and successful recovery does not establish that the model uses the information (Elazar et al., 2021; Belinkov, 2022). Causal tracing and representation interventions examine how changes to internal states affect model behavior (Meng et al., 2022; Geva et al., 2023; Wu et al., 2024), while model-based inspection can reveal information that direct vocabulary projection misses (Ghandeharioun et al., 2024). The interpretation of these interventions depends on their design, evaluation metrics, and controls (Zhang & Nanda, 2024; Elazar et al., 2021; Hase et al., 2023), and the best locations for editing knowledge need not match those identified by causal tracing (Hase et al., 2023). Related studies extend the analysis of factual knowledge to vision-language and speech-language models (Basu et al., 2024; Modica et al., 2026; Chung et al., 2026). Together, these studies show why different tests are needed to understand the relationship between internal information and generated answers. We apply these distinctions to failed speech recall, asking whether learned information remains present and contributes to the response, and whether strengthening the measured changes can improve the answer.

**Acquisition surface and update-induced interference.** Fine-tuning can teach a model new information while also affecting capabilities outside the training task (Luo et al., 2023; Gekhman et al., 2024; Cuervo et al., 2026). How that information is presented during training also affects what the model can later answer (Berglund et al., 2024; Zhao et al., 2025; Bhushan et al., 2025): question–answer pairs can support stronger knowledge generalization than document-style training data (Zhao et al., 2025). Speech adaptation can also weaken existing text capabilities (Cuervo et al., 2026; Wang et al., 2026b). These findings make it important to distinguish changes in overall model performance from the contribution of a particular learned fact. General interference may worsen speech responses even when newly learned information still contributes to them. Our interventions examine whether the learned answer affects a failed speech response without assuming that this contribution explains the entire speech–text performance gap. We also examine learning from declarative statements to assess how our findings depend on the way facts are taught, rather than assuming that statement-based and question–answer training produce equivalent results.

# 3 Characterizing Failed Cross-Modal Recall

Our goal is to understand whether knowledge learned from text still contributes to a model's response when it fails to answer a spoken question correctly. We first describe the controlled learning and evaluation setting in Section 3.1 and define the speech-update vector in Section 3.2. This vector is used to test whether the changes caused by text-based learning affect the response in Section 3.3 and reflect the specific answer the model learned in Section 3.4. Finally, in Section 3.5, we examine how these changes relate to those observed with text input and whether strengthening them or supplying information from text can improve the answer.

## 3.1 A controlled text-write, speech-query setting

To examine whether knowledge learned from text can support responses to spoken questions, we fine-tune a speech language model on text question–answer pairs and evaluate it on the same questions in both text and speech. This setting allows us to identify cases where the model answers a question correctly with text input but fails to answer it correctly with speech input.

For each factual item $i$, we denote the question by $q_i$, the answer by $a_i = (a_{i,1}, \ldots, a_{i,L_i})$, and the matched text and speech inputs by $x_{i,T}$ and $x_{i,S}$, respectively. We denote the base and text-fine-tuned models by $\mathcal{M}_{\boldsymbol{\theta}_0}$ and $\mathcal{M}_{\boldsymbol{\theta}_T}$, respectively.

We fine-tune the model on text question–answer pairs by minimizing the expected answer-token loss:

$$\mathcal{J}_T(\boldsymbol{\theta}) = -\mathbb{E}_{i\sim\mathcal{D}}\left[\frac{1}{|\mathcal{A}_i^T|}\sum_{t\in\mathcal{A}_i^T}\log p_{\boldsymbol{\theta}}(a_{i,t} \mid x_{i,T}, a_{i,<t})\right]. \tag{1}$$

Here, $i$ is sampled uniformly from the training set $\mathcal{D}$, and $\mathcal{A}_i^T$ denotes the answer-token positions included in the training loss. The corresponding speech question receives no example-level answer supervision during Text-FT. We also fine-tune the model on speech question–answer pairs (Speech-FT) as a behavioral control. The following analysis focuses on changes induced by text-only fine-tuning.

## 3.2 Measuring How Text-Based Learning Changes Speech Computation

To measure how text-only fine-tuning changes the model's response to a speech query, we compare its hidden states before and after fine-tuning:

$$\mathbf{u}_i = \mathbf{h}_{i,S,\ell}^{\mathrm{TextFT}} - \mathbf{h}_{i,S,\ell}^{\mathrm{Base}}. \tag{2}$$

We call $\mathbf{u}_i$ the speech-update vector. Here, $i$ identifies the question, $S$ denotes speech input, and $\ell$ is the prespecified late layer. We take both hidden states at the last token of the speech prompt, before the model starts answering. Both models receive the same speech waveform and prompt. This difference shows how text-only fine-tuning changes the hidden state at the selected layer. It may reflect changes beyond those related to the answer to this question. No answer string or text-route state is used to construct $\mathbf{u}_i$.

After fine-tuning, we keep all model parameters fixed and modify only the selected hidden state before the model starts answering. We evaluate questions that the unmodified TEXT-FT model answers correctly from text and divide them into two groups: **(1) speech successes**, which it also answers correctly from speech, and **(2) speech failures**, which it answers incorrectly from speech. Correctness is determined by exact match. We then analyze how removing the speech-update vector affects each group in Section 3.3.

### 3.3 DOES THE SPEECH-UPDATE VECTOR CONTRIBUTE TO ANSWERING?

To test whether the speech-update vector contributes to answering a spoken question, we remove it from the model's hidden state and measure the change in answer quality. We also remove a control vector from another question to test whether the effect is specific to the question being answered.

For each question, we compare three conditions. **(1) No removal** leaves the model unchanged. **(2) Same-question removal** subtracts the question's own speech-update vector $\mathbf{u}_i$. **(3) Matched control removal** subtracts a vector from a different author's question about the same relation. We rescale the control vector to have the same norm as $\mathbf{u}_i$, so both removals have the same magnitude. All other settings are held fixed. Same-question removal restores the BASE hidden state at the selected position, while the rest of the model remains TEXT-FT. In our experiments, we also use a random vector with the same norm as an additional control.

We use ROUGE-L recall to measure answer quality and compute two effects for each group $C$, either speech successes or speech failures:

$$D_C = \mathbb{E}_{i\sim C}\left[R_i(\text{no removal}) - R_i(\text{same-question removal})\right], \tag{3}$$

$$K_C = \mathbb{E}_{i\sim C}\left[R_i(\text{matched control removal}) - R_i(\text{same-question removal})\right]. \tag{4}$$

Here, $R_i$ is the recall score for question $i$ under the specified condition. **Recall drop** ($D_C$) measures how much answer quality decreases after same-question removal. **Question-specific effect** ($K_C$) measures how much more damaging same-question removal is than matched control removal. Positive values indicate that the vector contributes to answering and that its contribution is question-specific, respectively. These tests measure the effect of removing the vector, but do not establish that it is the only way the model represents or uses the learned information.

### 3.4 DOES THE SPEECH-UPDATE VECTOR REFLECT THE LEARNED ANSWER?

In this section, we test whether the speech-update vector reflects the specific answer learned during text fine-tuning. We train two models on the same questions but assign different answers to each model. The True model learns the original answers, while the Counterfactual model learns alternative answers. Both models start from the same base checkpoint and receive identical speech inputs during evaluation. For each model $c \in \{T, C\}$, we compute the speech-update vector as $\mathbf{u}_i^c = \mathbf{h}_i^c - \mathbf{h}_i^{\text{Base}}$.

Within each model, we subtract either its own vector or the other model's vector from the selected hidden state. For example, in the True model, we remove either the True vector or the Counterfactual vector. We rescale the other model's vector so that both removals have the same magnitude. We then compare the model's preference for its learned answer under the two removals. If the vector reflects the learned answer, removing the model's own vector should leave a weaker preference for that answer than removing the other model's vector.

To measure this preference, we score both candidate answers after each removal. The vector is constructed and removed before either answer is supplied. During scoring, we provide each answer's preceding tokens to predict its next token. We compute the negative log likelihood, averaged over the scored answer-token positions $\mathcal{A}_i(a)$:

$$\mathcal{L}_i^{c,z}(a) = -\mathbb{E}_{t\sim\mathcal{A}_i(a)}\left[\log p^z_{\boldsymbol{\theta}_c}\left(a_t \mid x_{i,S}, a_{<t}\right)\right], \tag{5}$$

where $z$ denotes the removal condition. Lower values mean that the model assigns a higher score to the candidate answer.

We measure preference for the learned answer by subtracting its loss from the alternative answer's loss:

$$P_i^c(z) = \mathcal{L}_i^{c,z}(a_i^{\neg c}) - \mathcal{L}_i^{c,z}(a_i^c), \tag{6}$$

where $a_i^c$ is the answer learned by model $c$, and $a_i^{\neg c}$ is the alternative. We call $P_i^c$ the **learned-answer log-probability margin**. A larger margin means stronger preference for the learned answer. Our **control-adjusted effect** compares the two removals:

$$\mathbb{E}_i \left[ P_i^c(\text{other-model removal}) - P_i^c(\text{own removal}) \right]. \tag{7}$$

A positive effect means that removing the model's own vector leaves a lower margin than removing the other model's vector. This is a comparison between the two removals, so neither margin needs to fall below the no-removal baseline. We report the effect for each model and their average.

We also test the two vectors in the same model. This checks whether their different effects persist when the model processing them is held fixed. We add either the True or Counterfactual vector to the same speech hidden state in the frozen base model, matching their norms in each comparison. We then compare preference for the True answer over the Counterfactual answer. A higher preference under the True vector shows that the two vectors have different effects consistent with their training answers. We separately compare each injection with no intervention to test whether each vector shifts preference toward its associated answer.

Finally, we ask whether the base model's output head can read answer information directly from the edited state. We add either vector to the same base-model speech state at equal strength, then apply the frozen RMSNorm and output head. For answer pairs with different first tokens, we compare the logits of those tokens. We repeat the comparison with an answer pair from another question about the same relation to check whether the effect is specific to the current question's answers. This test uses no answer continuation or fitted probe and measures only the first answer token. Full-answer generation is evaluated separately.

### 3.5 Can amplifying the speech-update vector improve answering?

In this section, we test whether strengthening the speech-update vector improves answers that the model originally gets wrong from speech. The removal tests measure whether the vector contributes to the response. Here, we ask whether increasing that contribution helps the model answer correctly.

We keep the TEXT-FT model fixed and scale the vector at the selected hidden state:

$$\mathbf{h}_i(\alpha) = \mathbf{h}_i^{\text{Base}} + \alpha \mathbf{u}_i. \tag{8}$$

Setting $\alpha = 0$ removes the vector, while $\alpha = 1$ restores the original TEXT-FT state. Values above one amplify the same vector at the same position. The rest of the model remains TEXT-FT in every condition. We measure mean ROUGE-L recall across strengths and compare amplification with the original state.

We next test whether a hidden state obtained from successful text recall can improve the same speech responses. We replace the selected speech state with the TEXT-FT model's text state for the same question. We compare this replacement with wrong and permuted text states under matched intervention conditions. This tests whether providing a successful text state helps where scaling the speech-update vector does not. The replacement supplies information from the text route, so its effect does not establish that the original speech state already contained that information.

Two additional analyses examine the update before intervention. First, we use cosine similarity to test whether a speech-update vector can identify the corresponding text-update vector among other questions' vectors. Second, we compare correct-answer and wrong-answer scores before and after text fine-tuning, without editing any hidden states. These tests measure whether the updates correspond across input routes and whether learning changes answer preference before intervention. Their detailed procedures, together with the intervention controls, appear in Appendices C–F.

## 4 Experimental Setup

We describe the models and data, question sets, intervention settings, and evaluation procedures used to characterize failed cross-modal recall.

### 4.1 MODELS, DATA, AND KNOWLEDGE ACQUISITION

We use Qwen2.5-Omni-7B (Xu et al., 2025) and MiMo-Audio-7B-Instruct (Xiaomi LLM-Core Team, 2025), both of which answer text and speech questions with text. TOFU (Maini et al., 2024) provides 4,000 question–answer pairs about 200 fictional authors. We synthesize each question with CosyVoice (Du et al., 2024), retaining the same reference answer. Within each model family, TEXT-FT and SPEECH-FT start from the same BASE checkpoint and learn the same answers through text and speech questions, respectively. We evaluate the final training checkpoints. Appendix A provides training and implementation details.

### 4.2 ANALYSIS POPULATIONS

To test answer specificity, we train two Qwen2.5-Omni-7B models on original or counterfactual answers to the same 796 questions about 40 authors. The remaining 3,204 answers, question wording, initialization, and training conditions are identical. We evaluate all 796 questions regardless of answer correctness. To study failed speech recall, we select 256 questions per model family that TEXT-FT answers exactly from text but incorrectly from speech. We exclude preliminary layer-selection questions and sample proportionally across four groups defined by pre-intervention speech ROUGE-L scores. For comparison with successful recall, we evaluate all 416 Qwen2.5-Omni-7B questions answered exactly through both routes.

### 4.3 INTERVENTION SCHEDULES AND REPLICATION SETTINGS

We apply the vector interventions from Section 3 at the final speech-prompt position, using layer 27 in Qwen2.5-Omni-7B and layer 33 in MiMo-Audio-7B-Instruct. Qwen2.5-Omni-7B uses cached decoding, so we edit the state once before answer generation. Our MiMo-Audio-7B-Instruct implementation recomputes the sequence at every generation step, requiring the same fixed vector to be reapplied at the same prompt position. We analyze the models separately. To test other authors, wording, and voices together, we repeat the Qwen2.5-Omni-7B evaluation on 280 questions from 14 previously excluded authors, using official paraphrases and a second CosyVoice voice without selecting for speech failure. In exploratory follow-ups, we add strengths $\alpha = 0.5, 1.5$ to Qwen's original $\alpha = 0, 1, 2$ and evaluate all five strengths on MiMo's separate failure cohort.

### 4.4 EVALUATION METRICS AND STATISTICAL PROCEDURES

We evaluate generated answers using ROUGE-L recall as the primary metric and exact match for complete reference recovery. Greedy decoding allows up to 160 new tokens for Qwen2.5-Omni-7B and 128 for the MiMo-Audio-7B-Instruct intervention experiments. Answer-preference analyses use mean answer-token negative log likelihood, with first-token loss as a secondary metric. Unless stated otherwise, tables report estimates with standard deviations from 10,000 bootstrap resamples. For the main intervention analyses, we resample whole authors, keeping their questions and paired intervention results together. Statistical decisions use analysis-specific bootstrap confidence intervals, including simultaneous intervals for designated comparisons. For $\alpha = 1.5$ and 2 versus 1, the equivalence margin is $\pm 0.03$ in mean ROUGE-L recall. Appendix G details the statistical procedures.

## 5 RESULTS

### 5.1 ANSWER SPECIFICITY OF THE SPEECH-UPDATE VECTOR

We compare the True and Counterfactual Qwen2.5-Omni-7B models to examine whether the speech-update vector reflects the answer learned during training. Table 1 shows that removing each model's own vector leaves a weaker preference for its learned answer than removing the other model's norm-matched vector. The differences are 0.0415 and 0.0630 for the True and Counterfactual models, respectively. Since the models learn different answers to the same questions, this comparison supports answer specificity beyond question identity.

We also compare the vectors within the same frozen BASE model. The True vector produces a stronger preference for the True answer than the Counterfactual vector. However, the Counterfactual

Table 1: **Answer preference after vector removal.** We compare removing the model's own speech-update vector with removing a norm-matched vector from the other model. Under both training targets, removing the model's own vector leaves a weaker preference for its learned answer.

| Training target | Own removal | Matched control | **Control-adjusted effect ↑** |
|---|---|---|---|
| True | 4.410 | 4.452 | **+0.042 [0.033, 0.051]** |
| Counterfactual | 2.947 | 3.010 | **+0.063 [0.054, 0.072]** |

Table 2: **Vector removal in successful and failed speech recall.** We compare vector removal on questions answered correctly or incorrectly from speech. Removing the question's own vector lowers ROUGE-L recall more than removing a norm-matched vector from another question in both groups, with larger effects on successful answers.

| Original speech answer | No removal | Same-question removal | Matched control removal | **Recall drop ↑** | **Question-specific effect ↑** |
|---|---|---|---|---|---|
| **Correct** | 1.000 | 0.817 | 0.926 | $\mathbf{0.183 \pm 0.015}$ | $\mathbf{0.109 \pm 0.016}$ |
| **Incorrect** | 0.453 | 0.382 | 0.425 | $\mathbf{0.071 \pm 0.014}$ | $\mathbf{0.043 \pm 0.015}$ |

vector does not increase preference for its associated answer relative to no intervention. The vectors therefore have distinguishable effects, but do not both steer the BASE model toward their associated answers. A frozen-head readout also detects answer-specific differences at the first answer token without an answer continuation (Appendices C.2–C.3).

### 5.2 CONTRIBUTION TO SUCCESSFUL AND FAILED SPEECH RECALL

We test whether the speech-update vector contributes to answering even when speech recall fails. On the selected Qwen2.5-Omni-7B questions, same-question removal lowers ROUGE-L recall more than norm-matched control removal in both the success and failure groups (Table 2). The recall drop from no intervention is 0.0707 for failures and 0.1830 for successes. The answer-likelihood analysis shows the same ordering on failures, including at the first answer token (Appendix C.4). These results show that the vector contributes to responses that are nevertheless incorrect. The larger drop on successes describes groups with different baseline answer quality and does not establish why their correctness differs.

### 5.3 REPLICATION ACROSS MODELS AND SPEECH CONDITIONS

We repeat the removal comparison on 280 Qwen2.5-Omni-7B questions with new authors, official paraphrases, and another synthesized voice. This set includes questions regardless of their original speech-answer correctness. Same-question removal causes more damage than norm-matched control removal, with a ROUGE-L difference of $0.0198 \pm 0.0078$. MiMo-Audio-7B-Instruct shows the same ordering, with a difference of $0.0475 \pm 0.0151$ under its repeated-edit schedule. The removal effect thus persists under the combined changes in the replication set and in a second model.

### 5.4 ALIGNMENT AND AMPLIFICATION OF SPEECH-UPDATE VECTORS

Using speech-update vectors, we retrieve the corresponding text-update vector for 209 of 256 Qwen failures at layer 27, compared with one at layer 4. Retrieval falls to 61 of 256 at Qwen's final layer, while MiMo maintains high retrieval in late layers (Figure 2(a)). These results establish correspondence between the updates across input routes, while the preceding removal tests establish their contribution to answering.

In both models, most mean gains occur as the removed vector is restored (Figure 2(b)). For Qwen, doubling the vector changes mean ROUGE-L recall by $+0.0010 \pm 0.0089$ relative to no intervention, and both tested amplification levels satisfy the equivalence criterion. For MiMo, the mean changes at strengths 1.5 and 2 are +0.0028 and $-0.0132$, respectively, and neither comparison satisfies that criterion. Replacing the speech state with the same question's successful text state

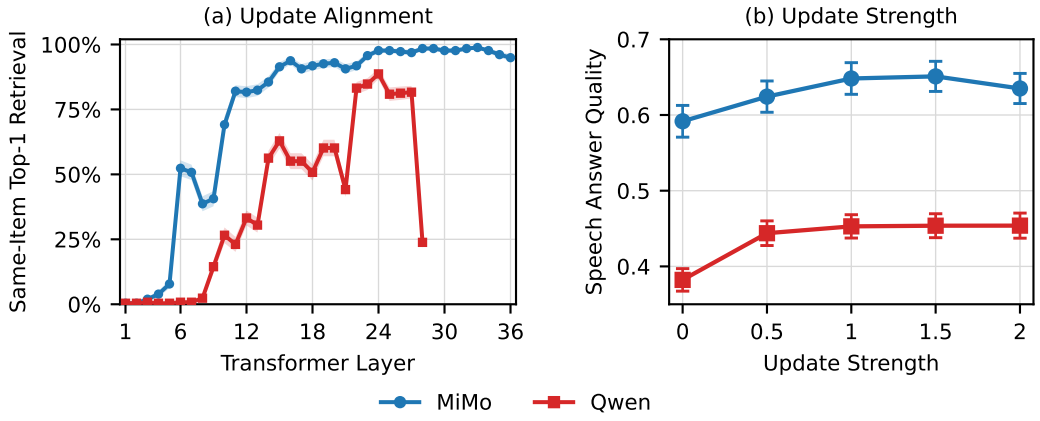



Figure 2: **Alignment and amplification of speech-update vectors.** (a) We retrieve the corresponding text-update vector among 256 candidates using speech-update vectors at each layer before final normalization. (b) We scale the speech-update vector in each fixed TEXT-FT model and measure ROUGE-L recall. Each model uses its own set of 256 speech failures. Strength zero removes the vector, one leaves the hidden state unchanged, and two doubles the vector.

produces larger improvements of 0.0885 for Qwen2.5-Omni-7B and 0.0831 for MiMo-Audio-7B-Instruct (Appendix E.5). This replacement introduces information from the text route.

## 6 Discussion and Conclusion

We investigate whether knowledge acquired through text still contributes to responses when speech recall fails. By comparing hidden states before and after text fine-tuning, we identify speech-update vectors whose effects reflect the learned answers. Removing these vectors worsens responses even on questions the model originally answers incorrectly. Amplifying them, however, yields little additional improvement in mean answer quality, whereas supplying a successful text state improves recall. These findings show that text-based learning can contribute to speech responses without being sufficient for correct answering.

Our experiments focus on Qwen2.5-Omni-7B and MiMo-Audio-7B-Instruct trained on question–answer pairs and evaluated with synthesized speech. Learning from declarative statements produces smaller effects that depend on question wording. The vector interventions identify the contribution of selected hidden-state changes, while the broader causes of the text–speech gap remain unresolved. In particular, the name recognition controls cover specific author contrasts, and other effects of fine-tuning remain possible contributors to the gap. Answer correctness, the contribution of a learned update, and the effectiveness of an intervention should therefore be evaluated separately.

### Reproducibility Statement

Appendices A–F specify interfaces, cohort construction, state extraction, controls, and intervention protocols. Appendix G gives resampling, protocol validation, and metric reconstruction.

### AI Use Statement

We used generative AI tools to polish the language of the manuscript. The authors are responsible for the final content.

## References

Junyi Ao, Yuancheng Wang, Xiaohai Tian, Dekun Chen, Jun Zhang, Lu Lu, Yuxuan Wang, Haizhou Li, and Zhizheng Wu. SD-Eval: A benchmark dataset for spoken dialogue understanding beyond

words. In *NeurIPS*, 2024.

Samyadeep Basu, Martin Grayson, Cecily Morrison, Besmira Nushi, Soheil Feizi, and Daniela Massiceti. Understanding information storage and transfer in multi-modal large language models. In *NeurIPS*, 2024.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Comput. Linguistics*, 48(1):207–219, 2022.

Lukas Berglund, Meg Tong, Maximilian Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. The reversal curse: LLMs trained on "A is B" fail to learn "B is A". In *ICLR*. OpenReview.net, 2024.

Kushagra Bhushan, Yatin Nandwani, Dinesh Khandelwal, Sonam Gupta, Gaurav Pandey, Dinesh Raghu, and Sachindra Joshi. Systematic knowledge injection into large language models via diverse augmentation for domain-specific RAG. In *NAACL (Findings)*, volume NAACL 2025 of *Findings of ACL*, pp. 5922–5943. Association for Computational Linguistics, 2025.

Michał Brzozowski, Zuzanna Dubanowska, Enrico Cassano, and Neo Christopher Chung. Reading the finetuning prior: Verbatim content recovery via contrastive decoding diffing. *CoRR*, abs/2605.25902, 2026.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen2-Audio technical report. *CoRR*, abs/2407.10759, 2024.

Sung Kyun Chung, Jiaheng Dong, Qiuchi Hu, Jinuo Sun, Gongping Huang, Hong Jia, and Ting Dang. Localizing and editing knowledge in large audio-language models. In *Interspeech*, pp. 6416–6420. ISCA, 2026.

Santiago Cuervo, Skyler Seto, Maureen de Seyssel, Richard He Bai, Zijin Gu, Tatiana Likhomanenko, Navdeep Jaitly, and Zakaria Aldeneh. Closing the gap between text and speech understanding in LLMs. In *ICLR*, 2026.

Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. Moshi: a speech-text foundation model for real-time dialogue. *CoRR*, abs/2410.00037, 2024.

Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, Zhifu Gao, and Zhijie Yan. CosyVoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens. *CoRR*, abs/2407.05407, 2024.

Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Trans. Assoc. Comput. Linguistics*, 9:160–175, 2021.

Shuhao Fu, Esther Goldberg, Ying Nian Wu, and Hongjing Lu. Multimodal function vectors for visual relations. In *ICML*, 2026.

Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. Does fine-tuning LLMs on new knowledge encourage hallucinations? In *EMNLP*, pp. 7765–7784. Association for Computational Linguistics, 2024.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In *EMNLP*, pp. 12216–12235. Association for Computational Linguistics, 2023.

Asma Ghandeharioun, Avi Caciularu, Adam Pearce, Lucas Dixon, and Mor Geva. Patchscopes: A unifying framework for inspecting hidden representations of language models. In *ICML*, volume 235 of *Proceedings of Machine Learning Research*, pp. 15466–15490. PMLR / OpenReview.net, 2024.

Peter Hase, Mohit Bansal, Been Kim, and Asma Ghandeharioun. Does localization inform editing? surprising differences in causality-based localization vs. knowledge editing in language models. In *NeurIPS*, 2023.

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

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *EMNLP/IJCNLP (1)*, pp. 2733–2743. Association for Computational Linguistics, 2019.

Ming-Hao Hsu, Xueyao Zhang, Xiaohai Tian, Jun Zhang, and Zhizheng Wu. Anatomy of the modality gap: Dissecting the internal states of end-to-end speech LLMs. *CoRR*, abs/2603.01502, 2026.

Jing Hu, Danxiang Zhu, Xianlong Luo, Dan Zhang, Shuwei He, Yishu Lei, Haitao Zheng, Shikun Feng, Jingzhou He, Yu Sun, Hua Wu, and Haifeng Wang. CORD: bridging the audio-text reasoning gap via weighted on-policy cross-modal distillation. *CoRR*, abs/2601.16547, 2026.

Brandon Huang, Chancharik Mitra, Assaf Arbelle, Leonid Karlinsky, Trevor Darrell, and Roei Herzig. Multimodal task vectors enable many-shot multimodal in-context learning. In *NeurIPS*, 2024.

Qian Jiang, Changyou Chen, Han Zhao, Liqun Chen, Qing Ping, Son Dinh Tran, Yi Xu, Belinda Zeng, and Trishul Chilimbi. Understanding and constructing latent modality structures in multimodal representation learning. In *CVPR*, pp. 7661–7671. IEEE, 2023.

Thomas Jiralerspong and Trenton Bricken. Cross-architecture model diffing with crosscoders: Unsupervised discovery of differences between LLMs. *CoRR*, abs/2602.11729, 2026.

Aly M. Kassem, Thomas Jiralerspong, Negar Rostamzadeh, and Golnoosh Farnadi. Delta-Crosscoder: Robust crosscoder model diffing in narrow fine-tuning regimes. *CoRR*, abs/2603.04426, 2026.

Kenneth Li, Oam Patel, Fernanda B. Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *NeurIPS*, 2023.

Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y. Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. In *NeurIPS*, 2022.

Jack Lindsey, Adly Templeton, Jonathan Marcus, Thomas Conerly, Joshua Batson, and Christopher Olah. Sparse crosscoders for cross-layer features and model diffing. Transformer Circuits Thread, 2024. URL `https://transformer-circuits.pub/2024/crosscoders/index.html`.

Grace Luo, Trevor Darrell, and Amir Bar. Vision-language models create cross-modal task representations. In *ICML*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *CoRR*, abs/2308.08747, 2023.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. TOFU: A task of fictitious unlearning for LLMs. In *COLM*, 2024.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *NeurIPS*, 2022.

Julian Minder, Clément Dumas, Stewart Slocum, Helena Casademunt, Cameron Holmes, Robert West, and Neel Nanda. Narrow finetuning leaves clearly readable traces in activation differences. *CoRR*, abs/2510.13900, 2025.

Luca Modica, Filip Landin, Mehrdad Farahani, Livia Qian, Gabriel Skantze, and Richard Johansson. Do factual recall mechanisms carry over from text to speech in multimodal language models? In **SEM*, pp. 401–409. Association for Computational Linguistics, 2026.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. LLMs know more than they show: On the intrinsic representation of LLM hallucinations. In *ICLR*. OpenReview.net, 2025.

Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. SALMONN: towards generic hearing abilities for large language models. In *ICLR*. OpenReview.net, 2024.

Eric Todd, Millicent L. Li, Arnab Sen Sharma, Aaron Mueller, Byron C. Wallace, and David Bau. Function vectors in large language models. In *ICLR*. OpenReview.net, 2024.

Chaoren Wang, Heng Lu, Xueyao Zhang, Shujie Liu, Yan Lu, Jinyu Li, and Zhizheng Wu. Closing the modality reasoning gap for speech large language models. In *ACL (1)*, pp. 18821–18835. Association for Computational Linguistics, 2026a.

Enzhi Wang, Qicheng Li, Zhiyuan Tang, and Yuhang Jia. Cross-modal knowledge distillation for speech large language models. In *ICASSP*, pp. 16112–16116. IEEE, 2026b.

Qiongqiong Wang, Hardik B. Sailor, Tianchi Liu, Wenyu Zhang, Muhammad Huzaifah, Nattadaporn Lertcheva, Shuo Sun, Nancy F. Chen, Jinyang Wu, and AiTi Aw. Benchmarking contextual and paralinguistic reasoning in speech-LLMs: A case study with in-the-wild data. In *EMNLP (Findings)*, pp. 14133–14148. Association for Computational Linguistics, 2025.

Yuxin Wen, Yangsibo Huang, Tom Goldstein, Ravi Kumar, Badih Ghazi, and Chiyuan Zhang. Quantifying cross-modality memorization in vision-language models. In *NeurIPS*, 2025.

Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D. Manning, and Christopher Potts. ReFT: Representation finetuning for language models. In *NeurIPS*, 2024.

Bajian Xiang, Shuaijiang Zhao, Tingwei Guo, and Wei Zou. Understanding the modality gap: An empirical study on the speech-text alignment mechanism of large speech language models. In *EMNLP*, pp. 5187–5202. Association for Computational Linguistics, 2025.

Xiaomi LLM-Core Team. MiMo-Audio: Audio language models are few-shot learners. *CoRR*, abs/2512.23808, 2025.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-Omni technical report. *CoRR*, abs/2503.20215, 2025.

Can Yaras, Siyi Chen, Peng Wang, and Qing Qu. Explaining and mitigating the modality gap in contrastive multimodal learning. In *CPAL*, volume 280 of *Proceedings of Machine Learning Research*, pp. 1365–1387. PMLR, 2025.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities. In *EMNLP (Findings)*, volume EMNLP 2023 of *Findings of ACL*, pp. 15757–15773. Association for Computational Linguistics, 2023.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *ICLR*. OpenReview.net, 2024.

Eric Zhao, Pranjal Awasthi, and Nika Haghtalab. From style to facts: Mapping the boundaries of knowledge injection with finetuning. In *NeurIPS*, 2025.

# A Experimental Setup and Intervention Definitions

We first describe how the models learn the controlled facts and how we modify their hidden states during evaluation. Appendix B defines the question sets, and Appendices C–F expand the comparisons in the main paper. Appendix G specifies the statistical procedures and checks used to validate the results. Unless stated otherwise, values after $\pm$ are bootstrap standard deviations as defined there.

## A.1 Training data and input formats

We train Qwen2.5-Omni-7B and MiMo-Audio-7B-Instruct on the same 4,000 TOFU question–answer pairs about fictional authors. Within each model family, Text-FT and Speech-FT use identical target answers and differ in whether the question is presented as text or speech. The fictional facts let us study learning from a controlled update, with conclusions limited to this setting.

For Qwen2.5-Omni-7B, each text training example starts with

```
Question: <question>
Answer:
```

followed by the target answer. Speech training and evaluation use the model's audio-chat format with no additional speech instruction. MiMo-Audio-7B-Instruct uses its official dialogue-SFT input formats with thinking disabled for both text and speech questions. Within each model family, checkpoint comparisons use identical prompts and speech waveforms.

## A.2 Training settings

We update the full Thinker in Qwen2.5-Omni-7B and use official full-model fine-tuning for MiMo-Audio-7B-Instruct. Table 3 gives the settings shared by text and speech training within each model family. The Qwen counterfactual model uses the same questions, data order, seed, and training settings as the True model. Only the answers to 796 questions differ, while the remaining 3,204 answers are unchanged. We evaluate the final checkpoints after ten epochs.

Table 3: **Training settings.** We use the same optimization settings for text and speech training within each model family.

| **Setting** | **Qwen2.5-Omni-7B** | **MiMo-Audio-7B-Instruct** |
|---|---|---|
| Trainable parameters | Full Thinker | Full model |
| Training examples | 4,000 | 4,000 |
| Epochs | 10 | 10 |
| Optimizer steps | 10,000 | 10,000 |
| Optimizer | AdamW | AdamW |
| Learning rate | $2 \times 10^{-5}$ | $2 \times 10^{-5}$ |
| Weight decay | 0 | 0 |
| Batch size | 1 | 1 |
| Gradient accumulation | 4 | 4 |
| Schedule / warmup | Cosine / 3% | Cosine / 3% |
| Gradient clipping norm | 1 | 1 |
| Checkpoint selection | Final epoch | Final epoch |

## A.3 Speech generation and answer scoring

The core experiments use speech synthesized by one English-female CosyVoice speaker at 16 kHz. We use greedy decoding with at most 160 new tokens for Qwen2.5-Omni-7B and 128 for the MiMo-Audio-7B-Instruct intervention experiments. MiMo generation stops at either the tokenizer's EOS token or the end-of-message token. We apply no sampling, repetition penalty, external language model, or post-hoc answer extraction.

To measure answer preference, we score each candidate answer after processing the speech prompt. Each candidate is scored in a separate forward pass using its preceding tokens to predict the next token. Only answer tokens contribute to mean negative log likelihood, with prompt, padding, EOS, and end-of-message positions excluded. First-token scoring uses the final prompt state before any candidate-answer token has been consumed.

### A.4 VECTOR REMOVAL AND AMPLIFICATION

We modify one hidden state to test the contribution of the speech-update vector defined in Section 3.2. For question $i$, let $\mathbf{h}_i^{\text{Base}}$ and $\mathbf{h}_i^{\text{TextFT}}$ be the states before and after text fine-tuning at the selected layer and final speech-prompt position. Both states are extracted before answer generation using the same prompt and waveform. Their difference is $\mathbf{u}_i = \mathbf{h}_i^{\text{TextFT}} - \mathbf{h}_i^{\text{Base}}$. All model parameters remain fixed during evaluation.

A control question $j$ concerns the same relation but a different author. We rescale its vector to match the length of the vector for question $i$:

$$\widetilde{\mathbf{u}}_{j\to i} = \frac{\|\mathbf{u}_i\|_2}{\|\mathbf{u}_j\|_2}\mathbf{u}_j. \tag{9}$$

The retained vectors have nonzero norm. This comparison holds edit magnitude fixed while changing which question supplied the vector. The resulting states for same-question removal, control removal, and amplification are

$$\mathbf{h}_i^{\text{same}} = \mathbf{h}_i^{\text{TextFT}} - \mathbf{u}_i = \mathbf{h}_i^{\text{Base}}, \tag{10}$$

$$\mathbf{h}_i^{\text{control}} = \mathbf{h}_i^{\text{TextFT}} - \widetilde{\mathbf{u}}_{j\to i}, \tag{11}$$

$$\mathbf{h}_i(\alpha) = \mathbf{h}_i^{\text{TextFT}} + (\alpha - 1)\mathbf{u}_i. \tag{12}$$

Thus, $\alpha = 0$ removes the vector, $\alpha = 1$ leaves the state unchanged, and $\alpha = 2$ doubles it. Restoring the Base state at this position leaves the remaining computation in the Text-FT model. It does not restore the entire Base model. For the opposing-answer comparison, we use the same normalization with the other checkpoint's vector for the same question.

Qwen2.5-Omni-7B uses cached decoding, so the edit is applied once during prompt processing. Our MiMo-Audio-7B-Instruct implementation recomputes the sequence at each generation step. It therefore reapplies the fixed vector to the newly computed state at the same prompt position on each forward pass. For MiMo, this position is the final group in its grouped text–audio representation. The vector and edited prompt position remain fixed throughout generation. We analyze the two schedules separately.

### A.5 TEXT-STATE REPLACEMENT AND MATCHED EDIT STRENGTH

We also test whether a state from successful text recall improves the response to a speech question. Let $\mathbf{t}_i$ be the Text-FT state at the same layer, taken from the final token of the text prompt for question $i$. Replacing the speech state with this text state applies the change

$$\mathbf{d}_i = \mathbf{t}_i - \mathbf{h}_i^{\text{TextFT}}, \qquad \mathbf{h}_i^{\text{replace}} = \mathbf{h}_i^{\text{TextFT}} + \mathbf{d}_i = \mathbf{t}_i. \tag{13}$$

A text state from another question $j$ instead produces the change $\mathbf{d}_{ij} = \mathbf{t}_j - \mathbf{h}_i^{\text{TextFT}}$. Both changes are nonzero, but their lengths can differ. The matched-strength comparison rescales the control change before applying it:

$$\widetilde{\mathbf{d}}_{ij} = \frac{\|\mathbf{d}_i\|_2}{\|\mathbf{d}_{ij}\|_2}\mathbf{d}_{ij}, \qquad \mathbf{h}_i^{\text{matched}} = \mathbf{h}_i^{\text{TextFT}} + \widetilde{\mathbf{d}}_{ij}. \tag{14}$$

The normalization matches the change to the speech state, rather than the norm of the text state itself. We compute this change relative to the live speech state before casting the edited state to the model's inference precision. Qwen applies the replacement once, while MiMo reapplies it at the fixed prompt position on every forward pass. Appendix E.5 specifies the text-state sources and controls. These replacements supply information from text processing and test whether that information can improve speech responses.

# B Recall Across Modalities and Question Selection

We first check whether the models can recover the learned answers from both written and spoken questions. We evaluate all 4,000 training questions under both input modalities (Table 4). Text fine-tuning produces near-perfect text recall in both model families, while speech recall remains lower. We use this difference to identify questions for the intervention studies.

Table 4: **Recall across input modalities.** We report mean ROUGE-L recall on all 4,000 questions. After text fine-tuning, both models recover the answers more reliably from text than from speech.

| Model | Training | Text query | Speech query |
|---|---|---|---|
| Qwen2.5-Omni-7B | TEXT-FT | **0.9994** | 0.5012 |
| | SPEECH-FT | 0.6487 | **0.8592** |
| MiMo-Audio-7B-Instruct | TEXT-FT | **0.9996** | 0.7516 |
| | SPEECH-FT | 0.9109 | **0.9674** |

## B.1 Selecting successful and failed speech recall

We select questions using the unmodified Text-FT model's answers, before observing any intervention result. A speech failure is answered exactly from text but incorrectly from speech. A speech success is answered exactly through both modalities. Let $e_{i,T}$ and $e_{i,S}$ indicate exact match for question $i$ under text and speech input, and let $\mathcal{E}$ contain questions not used for layer selection. The eligible failure and success sets are

$$\begin{aligned} \mathcal{F} &= \{i \in \mathcal{E} : e_{i,T} = 1,\ e_{i,S} = 0\}, \\ \mathcal{S} &= \{i \in \mathcal{E} : e_{i,T} = 1,\ e_{i,S} = 1\}. \end{aligned} \tag{15}$$

Table 5 gives the counts at each selection step for failures. We sample 256 failures per model, proportionally across four groups with pre-intervention speech ROUGE-L scores from 0 to 0.25, 0.25 to 0.5, 0.5 to 0.75, and 0.75 to 1. With the sampling seed fixed, Qwen2.5-Omni-7B contributes 68, 100, 39, and 49 questions from these groups, and MiMo-Audio-7B-Instruct contributes 36, 58, 32, and 130. We average intervention effects over the selected questions without population reweighting.

Table 5: **Selection of speech failures.** We retain questions answered correctly from text but incorrectly from speech, exclude layer-selection questions, and sample the remaining failures. Counts are cumulative.

| Selection step | Qwen2.5-Omni-7B | MiMo-Audio-7B-Instruct |
|---|---|---|
| All questions | 4,000 | 4,000 |
| Correct from text | 3,994 | 3,996 |
| Incorrect from speech | 3,578 | 2,721 |
| After layer-selection exclusions | 3,519 | 2,657 |
| Sampled failures | 256 | 256 |

For successful speech recall, we evaluate all 416 eligible Qwen2.5-Omni-7B questions from 138 authors. The success and failure groups have different initial answer quality and different room for a score to decrease. Their removal effects therefore describe how much each group depends on the tested vector, without identifying why the groups differ in correctness.

# C Answer Identity in Speech-Update Vectors

Does a speech-update vector reflect the answer learned during text fine-tuning? We test this in Qwen2.5-Omni-7B by changing the training answer while keeping the spoken question fixed. We then read the resulting vectors with a frozen output head and compare their effects in the same base model. These tests separate answer identity from differences in the question, vector norm, and model processing the edited state.

### C.1 CHANGING THE LEARNED ANSWER WHILE FIXING THE QUESTION

We evaluate all 796 questions from 40 authors whose training answers differ between the True-FT and Counterfactual-FT models. The models share the remaining training answers, optimization settings, data order, and seed. Evaluation uses identical questions and speech waveforms, without selecting questions by either model's answer quality.

At layer 27, each model receives one of three settings: no intervention, removal of its own speech-update vector, or removal of the other model's vector at the same norm. Both candidate answers are scored after the same edit to the final prompt state, before any answer token is supplied. We use the learned-answer preference $P_i^c$ defined in Equation 6 and compare other-vector removal with own-vector removal. The primary estimate averages this paired difference over questions and then over the two models. Table 6 reports both full-answer and first-token scores.

Table 6: **Answer preference after vector removal.** We report learned-answer preference after other-vector removal minus preference after own removal. Both scoring methods show lower preference after own removal.

| Answer scoring | True-FT | Counterfactual-FT | Average |
|---|---|---|---|
| Full answer (per token) | $0.0415 \pm 0.0036$ | $0.0630 \pm 0.0046$ | $\mathbf{0.0523} \pm 0.0040$ |
| First answer token | $1.292 \pm 0.080$ | $1.853 \pm 0.095$ | $\mathbf{1.573} \pm 0.084$ |

The paired effect is the preference under other-vector removal minus that under own-vector removal. For True-FT and Counterfactual-FT, mean full-answer preference is 4.5151 and 2.9470 without intervention, 4.4104 and 2.9474 after own removal, and 4.4519 and 3.0104 after other removal. Counterfactual-FT therefore gains 0.0004 under own removal and 0.0633 under other removal. Its positive paired effect reflects the smaller gain under own removal. All 40 leave-one-author-out estimates of the primary effect remain positive, ranging from 0.0501 to 0.0533.

### C.2 READING ANSWER IDENTITY WITH THE FROZEN OUTPUT HEAD

We next ask whether the base model's output head can distinguish the answers directly from the edited state. Of the 796 questions, 596 have different first tokens for the True and Counterfactual answers under the scoring tokenizer. For each question, we add either vector to the same Base speech state and match their norms. We repeat this comparison at both observed norms:

$$\mathbf{h}_i^{c,r} = \mathbf{h}_i^{\mathrm{Base}} + r\frac{\mathbf{u}_i^c}{\|\mathbf{u}_i^c\|_2}, \qquad c \in \{T, C\}, \quad r \in \{\|\mathbf{u}_i^T\|_2, \|\mathbf{u}_i^C\|_2\}. \tag{16}$$

The Base RMSNorm and output head map each edited state directly to token logits, without running the remaining transformer layers or supplying an answer continuation. No probe is fitted.

Let $g_i(\mathbf{h})$ be the True first-token logit minus the Counterfactual first-token logit for question $i$ under this readout. We also score an answer pair from a different author with the same relation, denoted by $g_j$. The control pairs follow a deterministic cycle within each relation, so no question receives its own pair. The answer-pair effect and its specificity are

$$A_i(r) = g_i(\mathbf{h}_i^{T,r}) - g_i(\mathbf{h}_i^{C,r}), \tag{17}$$

$$S_i(r) = A_i(r) - \left[g_j(\mathbf{h}_i^{T,r}) - g_j(\mathbf{h}_i^{C,r})\right]. \tag{18}$$

A positive $S_i(r)$ means that the vector directions distinguish the current question's answer pair more strongly than the control pair. Table 7 reports each norm separately and the average of the two effects. The average is taken after scoring both norms.

The mean answer-pair effect is 2.1794 for the current question and 0.1598 for the control pair. Answer-pair and specificity effects are positive for 78.5% and 72.5% of questions, respectively. All leave-one-author-out estimates remain positive for the six reported comparisons. This readout tests access to first-token information at the edited state. The following experiment tests the vectors through the remaining model computation.

Table 7: **Answer identity under a frozen output head.** We find that the vector directions distinguish the current question's answers more strongly than a control answer pair at both matched norms.

| Vector norm | Answer-pair effect | Specificity effect |
|---|---|---|
| True-vector norm | $2.1745 \pm 0.1195$ | $2.0170 \pm 0.1255$ |
| Counterfactual-vector norm | $2.1843 \pm 0.1203$ | $2.0221 \pm 0.1253$ |
| Average of both effects | $\mathbf{2.1794} \pm 0.1198$ | $\mathbf{2.0196} \pm 0.1253$ |

### C.3 Comparing vectors in a fixed base model

The removal experiment evaluates each vector in a fine-tuned model. To hold the model fixed, we instead inject either vector into the Base model for all 796 questions. We use the four edited states in Equation 16 and a no-intervention baseline, then score both complete answers through the remaining Base layers. An unscaled vector reconstructs its corresponding fine-tuned state at the edited position, while all model parameters remain those of Base.

For every setting, preference is the mean-token NLL of the Counterfactual answer minus that of the True answer. Table 8 first compares the two directions at each matched norm. It then compares each direction with no intervention, averaging the effects over the two norms.

Table 8: **Vector effects in a fixed base model.** We find higher True-answer preference with the True vector than with the Counterfactual vector, while the Counterfactual vector does not reduce that preference relative to no intervention.

| Comparison | Vector norm | Preference difference |
|---|---|---|
| True − Counterfactual | True | $0.0682 \pm 0.0052$ |
| | Counterfactual | $0.0687 \pm 0.0053$ |
| | Average of both effects | $\mathbf{0.0685} \pm 0.0053$ |
| True − no intervention | Average of both effects | $0.0854 \pm 0.0065$ |
| No intervention − Counterfactual | Average of both effects | $-0.0170 \pm 0.0047$ |

All 40 leave-one-author-out estimates remain positive for the True-minus-Counterfactual comparison at each norm and their average. The fixed model therefore responds differently to the two answer-associated directions. However, the comparison with no intervention does not support steering toward each vector's associated answer in both directions.

### C.4 Answer preference on selected speech failures

Does removing the speech-update vector reduce correct-answer preference even when the model originally answers incorrectly? We test the same 256 Qwen2.5-Omni-7B speech failures used for free generation. For each question, we score the correct answer against a wrong answer without intervention, after same-question removal, after matched control removal, and after removal of a deterministic random vector with the same norm. Every edit occurs once at the final prompt state before either candidate answer is supplied.

Table 9 compares matched control removal with same-question removal. For preference scores, we subtract the same-question result from the control result. For correct-answer NLL, we reverse the subtraction because a higher NLL means lower answer probability. Thus, positive values in all rows indicate greater damage from same-question removal.

Mean-token preference is 2.4114 without intervention, 2.1993 after same-question removal, 2.3659 after control removal, and 2.3412 after random removal. All 151 leave-one-author-out estimates of the primary preference difference remain positive. The first-token comparison shows that the effect is present before any candidate-answer token enters the model.

Table 9: **Answer preference on speech failures.** We find that same-question removal weakens correct-answer preference more than matched control removal, including at the first answer token.

| Scoring metric | Subtraction order | Difference |
|---|---|---|
| Mean-token preference | Control − same | $+\mathbf{0.1666} \pm 0.0086$ |
| First-token preference | Control − same | $+5.0030 \pm 0.2152$ |
| Correct-answer NLL | Same − control | $+0.0336 \pm 0.0053$ |

# D Vector Removal and Replication

The answer-scoring tests show that vector identity affects answer preference. Here, we measure its effect on freely generated responses. We compare removal on successful and failed speech recall, then repeat the test with new spoken questions and a second model family.

## D.1 Vector intervention settings

For Qwen2.5-Omni-7B, we edit the final speech-prompt state at layer 27 once, before generating the answer. Table 10 lists the settings using the vectors defined in Appendix A.4. The control vector comes from a different author's question about the same relation and is rescaled to the current question's vector norm. All settings share the Text-FT checkpoint, prompt, waveform, and greedy decoder. No text state or answer string is inserted. Appendix G describes the checks on state extraction and edit magnitude.

Table 10: **Vector intervention settings.** We remove or amplify the same-question vector and a matched control vector to test whether response changes depend on vector identity.

| Setting | Edited hidden state |
|---|---|
| No intervention | $\mathbf{h}_i^{\mathrm{TextFT}}$ |
| Same-question removal | $\mathbf{h}_i^{\mathrm{TextFT}} - \mathbf{u}_i$ |
| Matched control removal | $\mathbf{h}_i^{\mathrm{TextFT}} - \widetilde{\mathbf{u}}_{j \to i}$ |
| Same-question amplification | $\mathbf{h}_i^{\mathrm{TextFT}} + \mathbf{u}_i$ |
| Matched control amplification | $\mathbf{h}_i^{\mathrm{TextFT}} + \widetilde{\mathbf{u}}_{j \to i}$ |

## D.2 Removal on successful and failed speech recall

We compare the 256 selected failures with all 416 eligible successes defined in Appendix B. Table 11 reports paired differences in ROUGE-L recall. Recall drop $D$ compares no intervention with same-question removal, while the question-specific effect $K$ compares control removal with same-question removal. Both groups show a positive question-specific effect, with a larger recall drop on successes. The groups differ in initial correctness and available room for a score to decrease, so this comparison does not identify the cause of successful recall.

For failures, mean recall is 0.4529 without intervention, 0.3822 after same-question removal, 0.4252 after control removal, 0.4539 after same-question amplification, and 0.4319 after control amplification. For successes, recall is 1.0000, 0.8170, 0.9262, and 0.9747 under no intervention, same-question removal, control removal, and random removal, respectively. Exact match on successes follows the same order, with scores of 1.0000, 0.6226, 0.8413, and 0.9303. The random control has the same norm as the removed vector. A dash in Table 11 marks a comparison not reported there. All leave-one-author-out estimates of the primary group comparisons remain positive.

## D.3 Replication with new speech inputs and a second model

We first repeat the Qwen2.5-Omni-7B removal test on 280 questions from 14 new authors, covering all 20 relations. These questions use the official TOFU paraphrases and a second CosyVoice2 English voice. No question or author overlaps with the earlier Qwen analyses, and selection does

Table 11: **Removal effects on successful and failed recall.** We observe a larger recall drop on speech successes, while same-question removal is more damaging than matched control removal in both groups.

| Comparison | Speech failures | Speech successes |
|---|---|---|
| *Removal* | | |
| Control − same removal ($K$) | $+\mathbf{0.0431} \pm 0.0147$ | $+0.1092 \pm 0.0157$ |
| No intervention − same removal ($D$) | $+0.0707 \pm 0.0136$ | $+\mathbf{0.1830} \pm 0.0150$ |
| Random − same removal | — | $+0.1578 \pm 0.0159$ |
| *Amplification* | | |
| Same − control amplification | $+0.0220 \pm 0.0123$ | — |
| Same amplification − no intervention | $+0.0010 \pm 0.0089$ | — |
| Recall drop on successes − failures | $+\mathbf{0.1124} \pm 0.0198$ | |

not depend on answer quality. This test changes author, wording, and voice together, so it evaluates their combined change rather than their individual effects. The bootstrap seed for this replication is 3450.

We also test the 256 selected MiMo-Audio-7B-Instruct failures from 137 authors at layer 33. MiMo reapplies the fixed vector at the same prompt position on every generation forward pass, whereas Qwen applies it once. Table 12 therefore reports the models separately. The columns use the same question-specific effect $K$ and recall drop $D$ as above.

Table 12: **Replication of vector removal.** We find greater recall loss from same-question removal in both the new Qwen speech inputs and MiMo, using each model's intervention schedule.

| Model | Questions / authors | Question-specific effect ($K$) | Recall drop ($D$) |
|---|---|---|---|
| Qwen2.5-Omni-7B | 280 / 14 | $+0.0198 \pm 0.0078$ | $+0.0137 \pm 0.0068$ |
| MiMo-Audio-7B-Instruct | 256 / 137 | $+0.0475 \pm 0.0151$ | $+0.0565 \pm 0.0172$ |

All 14 Qwen and 137 MiMo leave-one-author-out estimates of $K$ remain positive. The Qwen effect is smaller than on the selected canonical failures but retains the same direction under the combined change in authors and speech inputs. MiMo shows the same ordering under its repeated-edit schedule.

# E Update Alignment and Intervention Effects

We now examine how speech-update vectors correspond to text-update vectors and whether increasing their strength improves failed answers. The retrieval analyses measure correspondence before intervention. The strength and text-state replacement experiments then test two ways of changing the response.

## E.1 Retrieving text updates from speech updates

We first evaluate Qwen2.5-Omni-7B on the same 256 selected speech failures. For each question and layer, we extract the Base and Text-FT states under text and speech input before either model receives an answer. The speech update $\mathbf{u}_{i,\ell}$ is defined as in Equation 2. Its text counterpart is $\mathbf{v}_{i,\ell} = \mathbf{h}^{\mathrm{TextFT}}_{i,T,\ell} - \mathbf{h}^{\mathrm{Base}}_{i,T,\ell}$. We compare these raw differences without centering, projection, or learned alignment.

For speech question $i$, let $\mathcal{C}_i$ be a set of candidate text questions that includes $i$. We rank candidates by cosine similarity and measure the separation from the strongest incorrect match:

$$s_{ij,\ell} = \frac{\mathbf{u}_{i,\ell}^\top \mathbf{v}_{j,\ell}}{\|\mathbf{u}_{i,\ell}\|_2 \|\mathbf{v}_{j,\ell}\|_2}, \tag{19}$$

$$m_{i,\ell} = s_{ii,\ell} - \max_{j \in \mathcal{C}_i \setminus \{i\}} s_{ij,\ell}. \tag{20}$$

Top-$k$ accuracy is the fraction of questions whose paired text update is among the $k$ highest-scoring candidates. The chance top-1 rate for question $i$ is $1/|\mathcal{C}_i|$. We average this rate across questions and subtract it from top-1 accuracy for chance-adjusted results. A positive cosine margin means that the correct text update ranks above every other candidate.

Table 13 compares unrestricted retrieval with candidates limited to the same author or relation. Unrestricted retrieval uses all 256 questions. The within-author comparison retains 177 questions with at least one other selected fact about the same author. Within-relation retrieval uses six to eighteen candidates per question.

Table 13: **Text-update retrieval from speech updates.** We find strong same-question retrieval at layer 27 of Qwen2.5-Omni-7B, including when candidates share the author or relation.

| Layer | Candidate pool | Questions | Chance | Top-1 | Cosine margin |
|---|---|---|---|---|---|
| 4 | All | 256 | 0.0039 | $0.0039 \pm 0.0039$ | $-0.0464 \pm 0.0011$ |
| 27 | All | 256 | 0.0039 | $0.8164 \pm 0.0240$ | $+0.0436 \pm 0.0031$ |
| 27 | Same author | 177 | 0.4068 | $0.9774 \pm 0.0110$ | $+0.1209 \pm 0.0055$ |
| 27 | Same relation | 256 | 0.0781 | $0.9453 \pm 0.0138$ | $+0.0919 \pm 0.0039$ |

Layer 27 retrieves 209 of 256 paired updates from the full candidate set, compared with one at layer 4. The prespecified retrieval comparisons pass the tests based on whole-author resampling and fixed target permutations. The updates therefore correspond across input modalities on these questions, even though the model fails to answer them correctly from speech.

We extend the analysis to all 28 Qwen layers in an exploratory follow-up, retaining the same 256 questions from 151 authors, checkpoints, and inputs. For Figure 2(a), we extract decoder-block outputs before final normalization at the final attended prompt position. These states exactly reproduce the original layer-4 and layer-27 states. Retrieval falls from 209 of 256 questions at layer 27 to 61 at layer 28, so correspondence does not increase monotonically with depth.

### E.2 Retrieval across question paraphrases

Retrieval from identically worded questions could depend on their shared wording. We therefore repeat the Qwen analysis using official TOFU paraphrases of the selected failures. After excluding paraphrases that change the requested fact, 49 questions remain from 26 authors and 20 relations. This selection is made before examining retrieval outcomes for the paraphrases.

The primary comparison uses paraphrased speech to retrieve canonical text updates. A positive control uses the same paraphrase for both speech and text. Each candidate set contains all 49 questions. Table 14 reports successful retrieval counts, with the expected rates under uniform chance in the final row.

Table 14: **Retrieval across paraphrases.** We retrieve the corresponding text update from paraphrased speech at layer 27 even when the text candidates retain the original wording.

| Text candidates | Layer 4 top-1 | Layer 27 top-1 | Layer 27 top-5 |
|---|---|---|---|
| Canonical | 1/49 | 32/49 | 43/49 |
| Paraphrased | 3/49 | 46/49 | 48/49 |
| Uniform chance | 1/49 | 1/49 | 5/49 |

With canonical text candidates, layer 27 retrieves the correct update for 32 of 49 questions at top-1 and 43 at top-5, with median rank one. Retrieval within author and relation also exceeds the

corresponding group-preserving permutation controls. Retrieval is stronger when both modalities use the same paraphrase, so we report this control separately. The primary result rules out identical wording as the sole explanation, while leaving question identity and shared semantics as possible sources of correspondence.

### E.3 Retrieval across MiMo layers

We next test whether the early-to-late retrieval pattern also appears in MiMo-Audio-7B-Instruct. We extract final prompt states from all 36 decoder blocks before final normalization for its 256 selected speech failures from 137 authors, using the official text and speech input formats without reference answers. At each layer, every speech update retrieves among all 256 text updates using the same cosine criterion.

The primary comparison subtracts mean top-1 accuracy over layers 1–10 from the mean over layers 29–34. Both layer bands are fixed before evaluation. We also compare layers 33 and 4, restrict candidates by author or relation, and repeat the band comparison after excluding all authors used in the layer-selection pilot. This last comparison retains 180 questions from 96 authors.

Table 15: **Text-update retrieval across MiMo layers.** We find stronger retrieval in late layers, with a similar late-minus-early difference after excluding authors used for layer selection.

| **Comparison** | **Evaluated questions** | **Top-1 difference** |
|---|---|---|
| Late band − early band | All 256 | $+0.7151 \pm 0.0135$ |
| Late band − early band | 180, pilot authors excluded | $+0.7094 \pm 0.0141$ |
| Layer 33 − layer 4 | All 256 | $+0.9492 \pm 0.0130$ |
| **Retrieval level** | **Evaluated questions** | **Adjusted top-1** |
| Late band, chance adjusted | All 256 | $+0.9772 \pm 0.0066$ |

Table 15 shows a positive late-minus-early difference in both question sets. Raw top-1 retrieval rises from 10 of 256 questions at layer 4 to 253 of 256 at layer 33. At layer 33, retrieval is perfect within author and succeeds for 255 of 256 questions within relation. The late-layer pattern therefore also includes MiMo's independently selected intervention layer, while the selected layer itself differs between models.

### E.4 Response to vector strength

We test five vector strengths on each model's 256 selected speech failures, comprising 151 authors for Qwen2.5-Omni-7B and 137 for MiMo-Audio-7B-Instruct. The edit in Equation 12 removes the vector at $\alpha = 0$, leaves the original state at $\alpha = 1$, and doubles the vector at $\alpha = 2$. Within each model, all five strengths share the questions, layer, checkpoint, waveforms, and decoding schedule.

For Qwen, we fixed the exploratory strengths 0.5 and 1.5 after observing the original strengths 0, 1, and 2. We extend this comparison to MiMo after observing the Qwen results, making the model extension exploratory. For MiMo, we retain the original outputs at strengths 0 and 1 and generate new outputs at 0.5, 1.5, and 2. Before the new generations, replaying strengths 0 and 1 on three fixed questions reproduces all six original outputs exactly. We apply the fixed vector at layer 33 under MiMo's repeated-edit schedule from Section 4.3.

Let $\bar{R}(\alpha) = n^{-1} \sum_{i=1}^{n} R_i(\alpha)$ be mean ROUGE-L recall at strength $\alpha$. We compare the gain from restoring the removed vector with the gain from doubling it:

$$C = \left[\bar{R}(1) - \bar{R}(0)\right] - \left[\bar{R}(2) - \bar{R}(1)\right]. \tag{21}$$

A positive $C$ indicates that the first increase in strength gives a larger mean gain than the second. For each model, we test whether the changes from 1 to 1.5 and from 1 to 2 meet the same equivalence rule. Both simultaneous confidence intervals must lie entirely within $(-0.03, 0.03)$, as detailed in Appendix G.

Table 16 shows that recall rises mainly as the removed vector is restored. For Qwen, both amplification comparisons satisfy the simultaneous equivalence criterion, supporting a plateau in mean

Table 16: **Response to vector strength.** We compare five strengths on each model's 256 selected speech failures. Mean recall rises mainly as the removed vector is restored. The lower rows report within-model differences.

| Vector strength ($\alpha$) | Qwen2.5-Omni-7B | MiMo-Audio-7B-Instruct |
|---|---|---|
| | Mean ROUGE-L recall | |
| 0 | $0.3822 \pm 0.0149$ | $0.5917 \pm 0.0210$ |
| 0.5 | $0.4438 \pm 0.0163$ | $0.6243 \pm 0.0207$ |
| 1 | $0.4529 \pm 0.0154$ | $0.6482 \pm 0.0209$ |
| 1.5 | $0.4538 \pm 0.0158$ | $0.6510 \pm 0.0199$ |
| 2 | $0.4539 \pm 0.0166$ | $0.6350 \pm 0.0199$ |
| **Comparison** | **Recall difference** | |
| $0.5 - 0$ | $+\mathbf{0.0617} \pm 0.0135$ | $+\mathbf{0.0326} \pm 0.0139$ |
| Concavity ($C$) | $+\mathbf{0.0697} \pm 0.0172$ | $+\mathbf{0.0697} \pm 0.0251$ |
| $1.5 - 1$ | $+0.0009 \pm 0.0064$ | $+0.0028 \pm 0.0093$ |
| $2 - 1$ | $+0.0010 \pm 0.0089$ | $-0.0132 \pm 0.0141$ |

recall along the tested direction over the observed strength range. For MiMo, the 95% simultaneous intervals for strengths 1.5 and 2 relative to 1 are $[-0.0250, 0.0306]$ and $[-0.0410, 0.0146]$, respectively. Both intervals include zero and extend beyond an equivalence boundary, establishing neither a plateau nor a clear mean improvement or decline.

### E.5 Replacing the speech state with a text state

We next test whether a state obtained during successful text recall can improve the same failed speech responses. We replace the selected speech state with the text state for the same question, using the operation in Appendix A.5. Qwen2.5-Omni-7B receives one replacement at layer 27 before generation. MiMo-Audio-7B-Instruct receives repeated replacements at the fixed prompt position at layer 33. Table 17 lists the controls for the supplied question, edit magnitude, replacement code, and layer. All settings share the question, waveform, prompt, checkpoint, and greedy decoder within each model.

Table 17: **Text-state replacement settings.** We compare the same-question text state with control states, noise, and an early-layer replacement to test which information improves speech responses.

| Setting | Source or edit | Layer |
|---|---|---|
| Same-question text | Text state for the current question | Late |
| Matched control text | Different author, same relation, closest answer length | Late |
| Permuted text | Text states shuffled with a fixed seed | Late |
| No intervention | Install the hook and leave the state unchanged | Late |
| Norm-matched noise | Add fixed random noise with the same norm as the same-question edit | Late |
| Early replacement | Text state for the current question | Early |

Qwen's early and late layers are fixed using a 61-question pilot. MiMo's layer 33 is selected using a 64-question pilot, with layer 4 fixed as the early control. All pilot questions are excluded from the corresponding evaluation sets. MiMo's exclusion is by question rather than author, leaving 76 selected questions whose authors also appear in the pilot. A later, non-prespecified analysis of the remaining 180 questions also gives a positive effect and is treated as supporting evidence.

The original control, permuted, and early replacements can change the speech state by different amounts. We therefore run a separate comparison that rescales the control and permuted edits to match the same-question edit norm, as defined in Equation 14. These follow-up settings are fixed before evaluating their outcomes. We retain the same questions, text-state sources, checkpoints,

inputs, layers, intervention schedules, and decoder. Table 18 reports same-question replacement minus each matched-strength control in ROUGE-L recall.

Table 18: **Text-state replacement at matched strength.** We report recall gains of same-question text states over each control at matched edit strength.

| Control | Qwen2.5-Omni-7B | MiMo-Audio-7B-Instruct |
|---|---|---|
| Matched control text | $+0.1982 \pm 0.0167$ | $+0.2394 \pm 0.0221$ |
| Permuted text | $+0.2105 \pm 0.0178$ | $+0.1873 \pm 0.0231$ |

All four matched-strength differences remain positive under leave-one-author-out deletion. To check whether the original replacement benefit depends on the answer metric, we compare same-question replacement with no intervention on the same 256 questions per model (Table 19). We report paired differences across answer-quality and output-length measures. This table uses 10,000 item-bootstrap resamples, rather than the author resampling used for Table 18.

Table 19: **Answer quality after text-state replacement.** We report gains over no intervention: answer quality improves across metrics, with little change in output length.

| Metric | Qwen2.5-Omni-7B | MiMo-Audio-7B-Instruct |
|---|---|---|
| *Answer quality* | | |
| Exact match | $+0.0742 \pm 0.0164$ | $+0.1563 \pm 0.0230$ |
| Token F1 | $+0.0800 \pm 0.0143$ | $+0.0735 \pm 0.0143$ |
| ROUGE-L precision | $+0.0876 \pm 0.0157$ | $+0.0776 \pm 0.0157$ |
| ROUGE-L recall | $+0.0885 \pm 0.0155$ | $+0.0831 \pm 0.0156$ |
| ROUGE-L F1 | $+0.0907 \pm 0.0156$ | $+0.0807 \pm 0.0157$ |
| *Output length* | | |
| Normalized output tokens | $+0.12 \pm 0.44$ | $+0.30 \pm 0.23$ |
| Output/reference token ratio | $-0.004 \pm 0.018$ | $+0.000 \pm 0.011$ |

A text state can therefore improve responses that show little mean benefit from amplifying the speech-update vector. The replacement supplies information from a separate text computation, so it does not establish that the original speech computation already used that information.

# F Supporting Controls and Limits

We examine three possible limits to the main interpretation in Qwen2.5-Omni-7B. The first test checks whether the model can distinguish the fictional authors' names. The second measures answer preference before intervention. The third replaces question–answer training with factual-statement training to test how far the observed pattern extends.

## F.1 Recognizing names in paired comparisons

We test name recognition using one question per author and 74 disjoint author pairs, giving 148 speech waveforms. For authors A and B, we score both names after the question about A and reverse their roles after the question about B. We repeat the comparison with written questions. Only the name's continuation tokens enter mean NLL. The preference margin is the other author's NLL minus the correct author's NLL, so positive values favor the name in the question.

We count a query as correct when this margin is positive and a pair as correct when both reciprocal queries are correct (Table 20). We resample complete pairs 10,000 times to preserve their dependence. This evaluates discrimination between two candidate names rather than open-ended transcription.

The Text-FT speech margin is also positive at the first name token, before any earlier name token has been supplied. Text input gives a larger Text-FT margin than speech input. The change in the speech margin from Base to Text-FT is unresolved under the simultaneous comparison rule. Open-ended

Table 20: **Paired name recognition.** We find that both checkpoints usually prefer the correct fictional author name under text and speech input.

| Checkpoint | Input | Preference margin | Correct queries | Correct pairs |
|---|---|---|---|---|
| BASE | Text | $+5.8728 \pm 0.1395$ | 147/148 | 73/74 |
| BASE | Speech | $+4.4992 \pm 0.1792$ | 144/148 | 70/74 |
| TEXT-FT | Text | $+7.8035 \pm 0.2375$ | 147/148 | 73/74 |
| TEXT-FT | Speech | $+\mathbf{4.9758} \pm 0.2314$ | 140/148 | 66/74 |

transcription did not meet the predefined transcript-quality criteria, so the evidence here remains limited to paired name recognition.

### F.2 ANSWER PREFERENCE BEFORE INTERVENTION

We ask whether text fine-tuning increases correct-answer preference on speech failures before any hidden-state edit. The analysis uses the same 256 selected Qwen2.5-Omni-7B failures. For each question, a fixed-seed permutation supplies a wrong answer from a different author with the same TOFU relation. No question receives its own answer, and the full set of wrong answers contains exactly the same answers as the correct-answer set. Question order, candidate pairs, inputs, checkpoints, scoring, and statistical comparisons are fixed before evaluation.

We score the correct and wrong answers separately under Base and Text-FT. Preference is the wrong-answer NLL minus the correct-answer NLL, with no state replacement or vector edit. Table 21 reports how much this preference increases from Base to Text-FT. The mean-token speech comparison is primary, and the other comparisons use simultaneous inference as specified in Appendix G.

Table 21: **Answer preference before intervention.** We report the increase in correct-over-wrong answer preference after text fine-tuning, including for failed speech queries.

| Answer scoring | Text query | Speech query |
|---|---|---|
| Mean token | $+2.0886 \pm 0.0734$ | $+\mathbf{1.1185} \pm 0.0682$ |
| First token | $+19.0949 \pm 0.4332$ | $+6.9002 \pm 0.2881$ |

The primary speech effect is positive for 219 of 256 questions and under every leave-one-author-out omission. Correct-answer mean-token NLL decreases by 2.6096, while wrong-answer NLL also decreases by 1.4911. At the first token, correct-answer NLL decreases by 1.2025 and wrong-answer NLL increases by 5.6977. Text fine-tuning therefore shifts preference toward the correct candidate on average, even when free generation does not recover it.

### F.3 LEARNING FROM FACTUAL STATEMENTS

The main experiments teach answers through question–answer pairs. We test a broader training format by fine-tuning Qwen2.5-Omni-7B on 796 factual statements about 40 authors, without including the questions. The true-statement model learns normalized versions of TOFU's official paraphrased answers. A control model instead learns the corresponding content-altered statements from TOFU's perturbed answers. We compare both models with Base using canonical and officially paraphrased questions, none of which appear in statement training.

Table 22 reports ROUGE-L recall for the true-statement model minus each reference checkpoint. The effects are small and vary with question wording. This analysis uses paired item-bootstrap resampling, as specified in Appendix G.

Speech recall remains below text recall for all three checkpoints under both question wordings, and exact match is at most 0.0063. Statement training yields much weaker text recall than direct question–answer training in these experiments. The two training settings differ in target length, supervised token positions, total token exposure, and realized optimization trajectory. These differences leave the role of update-induced interference unresolved.

Table 22: **Recall after statement training.** We compare true-statement training with each reference checkpoint. Recall changes are small and larger for paraphrased questions.

| Question wording | Reference checkpoint | Text | Speech |
|---|---|---|---|
| Canonical | BASE | $-0.0051 \pm 0.0061$ | $-0.0113 \pm 0.0057$ |
| | Content-altered control | $+0.0129 \pm 0.0049$ | $-0.0103 \pm 0.0049$ |
| Paraphrased | BASE | $+0.0194 \pm 0.0062$ | $+0.0264 \pm 0.0061$ |
| | Content-altered control | $+0.0301 \pm 0.0055$ | $+0.0295 \pm 0.0053$ |

# G STATISTICAL INFERENCE AND VALIDATION

We estimate intervention effects from paired outcomes on the same questions. This section defines how those effects are averaged, how uncertainty is computed, and which records are checked before analysis.

## G.1 PAIRED ESTIMATES AND RESAMPLING

For a comparison of settings $z_1$ and $z_0$, let $\Delta_i = y_i(z_1) - y_i(z_0)$ be the paired difference for question $i$. The score $y_i$ is recall, answer preference, or another metric specified for that comparison. Our estimate gives equal weight to each selected question:

$$\widehat{\Delta} = \frac{1}{n} \sum_{i=1}^{n} \Delta_i. \tag{22}$$

This estimates the mean effect on the selected questions, without reweighting to the full question population. For a single-setting mean, we use $y_i$ in place of $\Delta_i$.

Most analyses resample whole authors to retain dependence among their questions. Let $I_a$ contain the selected questions for author $a$, with $n_a = |I_a|$, and let $G$ be the number of authors. For bootstrap replicate $b$, we draw $G$ authors uniformly with replacement and denote the number of draws of author $a$ by $m_a^{(b)}$. We retain all questions and paired outcomes for each draw, giving

$$\widehat{\Delta}^{*(b)} = \frac{\sum_{a=1}^{G} m_a^{(b)} \sum_{i \in I_a} \Delta_i}{\sum_{a=1}^{G} m_a^{(b)} n_a}. \tag{23}$$

Authors with more selected questions still contribute more observations. This procedure does not replace the question-weighted estimate with an average of author means.

Unless stated otherwise, we use $B = 10{,}000$ resamples and seed 3407. The independent Qwen replication uses seed 3450. For the full-layer retrieval profiles, we use seed 3411 for the simultaneous intervals and plotted standard deviations. The values after $\pm$ are the sample standard deviation of the bootstrap estimates:

$$\mathrm{SD}_{\mathrm{boot}} = \sqrt{\frac{1}{B-1} \sum_{b=1}^{B} \left( \widehat{\Delta}^{*(b)} - \bar{\Delta}^{*} \right)^2}, \qquad \bar{\Delta}^{*} = \frac{1}{B} \sum_{b=1}^{B} \widehat{\Delta}^{*(b)}. \tag{24}$$

This measures uncertainty in the estimated mean, rather than variation across individual answers or training seeds. Tables 19 and 22 instead resample individual questions with their paired outcomes. Table 20 resamples complete reciprocal name pairs.

## G.2 CONFIDENCE INTERVALS AND COMPARISON FAMILIES

We retain the decision rule fixed for each analysis. Individual percentile intervals use the 2.5th and 97.5th percentiles of the bootstrap estimates. For simultaneous comparisons, all effects in a predefined family use the same resampling draws. If $\mathcal{J}$ indexes that family, its common interval half-width is

$$q_{0.95} = \mathrm{Quantile}_{0.95} \left( \left\{ \max_{j \in \mathcal{J}} \left| \widehat{\Delta}_j^{*(b)} - \widehat{\Delta}_j \right| \right\}_{b=1}^{B} \right). \tag{25}$$

The interval for effect $j$ is $[\widehat{\Delta}_j - q_{0.95}, \widehat{\Delta}_j + q_{0.95}]$. For the amplification plateau, both intervals must lie strictly within $(-0.03, 0.03)$. We use confidence intervals for statistical decisions and report bootstrap standard deviations to summarize uncertainty. Leave-one-author-out checks recompute the estimate after deleting all selected questions for one author at a time.

We analyze models separately and use individual intervals for primary effects unless a simultaneous rule is specified. The simultaneous families include checkpoint-specific opposing-answer effects, norm-specific frozen-head effects, and separate norm and no-intervention comparisons in the fixed Base model. Answer-pair and specificity readouts use separate families. For the original Qwen failed-recall study, removal and amplification comparisons share a family, while successful recall uses a family of matched and random controls. Each five-strength study separately groups the differences between strengths 1.5 and 1 and between strengths 2 and 1 into a two-comparison family. Each full-layer retrieval profile uses its own family of top-1 rates, spanning all 28 Qwen layers or all 36 MiMo layers. Secondary answer-preference and name-recognition scores are tested within their respective analyses. Text-state replacement uses four original controls, with a separate two-control family per model for the matched-strength comparison.

### G.3 Record, state, and metric validation

We check question coverage, paired settings, and model identity before recomputing answer metrics from retained outputs. Invalid or incomplete records fail validation rather than becoming zero-filled observations. For Qwen, stored Text-FT states match live generation prefills. Same-question removal reconstructs the paired Base state at the first edited forward pass in both models, exactly at the stored precision for Qwen.

The Qwen behavior evaluation reruns two questions whose original vocabulary records were undefined under the final tokenizer, retaining the other 3,998 outputs. Records from different seeds or prompts are not combined. Configurations, selected questions, outputs, and states are retained so that reported comparisons can be recomputed without rerunning inference.