DO CONTEXT-RELIANCE DIRECTIONS TRANSFER TO HALLUCINATION DETECTION IN RAG?

Anonymous authors
Paper under double-blind review

ABSTRACT

Representation engineering identifies directions in transformer activation space that can be used to measure or influence model behavior at inference time. In Retrieval-Augmented Generation (RAG), such a direction would be an attractive hallucination signal: it is intrinsic, costs one dot product per token, and uses information that post-hoc detectors discard by treating the generator as a black box. We ask whether *context reliance* is encoded as a single, transferable linear direction. We construct 15 contrastive datasets spanning hallucination-prone scenarios — 13 synthetic A/B templates, an external counterfactual QA corpus, and a template-free dialog construction — and extract one Context-Source Vector (CSV) per dataset from Llama-3.2-1B-Instruct and Llama-3.1-8B-Instruct. Each CSV is then scored on RAGTruth against two controls: 100 random directions (lower control) and a CSV fitted in-domain on RAGTruth (upper control). The answer is negative in an informative way. CSVs do transfer above chance, but only modestly: best-layer AUROC is 0.56–0.68 (mean 0.62) for the 1B model and 0.59–0.70 (mean 0.65) for the 8B model, against random-direction means of 0.57 and 0.58 and in-domain upper controls of 0.77 and 0.84. Crucially, CSVs that are nearly orthogonal or even negatively aligned with one another achieve indistinguishable detection performance, yet often peak at entirely different layers. Effective context-reliance signals are therefore not uniquely represented by a single activation-space direction; instead, the extracted directions appear to capture distinct, partly confounded features that reside at different depths in the network and are similarly predictive downstream. The larger model encodes more of these features, yielding a small but consistent improvement in transfer.

1 INTRODUCTION

Retrieval-Augmented Generation (RAG) (Lewis et al., 2020; Fan et al., 2024) has become the predominant paradigm for grounding large language model (LLM) outputs in verifiable external knowledge. By conditioning generation on retrieved context, RAG systems mitigate the tendency of LLMs to produce plausible but unfaithful content (Huang et al., 2025; Ji et al., 2023). Nevertheless, hallucinations persist even in RAG-augmented systems (Shuster et al., 2021; Das et al., 2026), i.e., models may disregard the retrieved context entirely and rely either on potentially outdated or erroneous parametric knowledge (Longpre et al., 2021) or fabricate information that appears in neither the context nor the training data. In high-stakes domains, the consequences of such failures are severe—exemplified by documented cases of AI-generated legal citations to non-existent case law (Weiser, 2023)—underscoring the critical need for reliable hallucination detection mechanisms.

Existing approaches to hallucination detection operate predominantly as *post-hoc* verification systems. These include consistency-based methods such as SelfCheckGPT (Manakul et al., 2023), natural language inference (NLI)-based faithfulness classifiers (Honovich et al., 2022), and uncertainty estimation techniques (Kadavath et al., 2022). While useful, such methods treat the generative model as a black box, discarding internal signals in the activation space available during inference. Recent advances in mechanistic interpretability (Elhage et al., 2021) and representation engineering (Zou et al., 2023) have demonstrated that high-level semantic concepts, including truthfulness (Li et al., 2023; Marks & Tegmark, 2024) and factual recall (Meng et al., 2022; Geva et al., 2021) are often encoded as linearly separable directions in a transformer's activation space.

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.

Although activation-space approaches have begun to target context faithfulness, it remains unclear whether *context reliance* in RAG is encoded as a transferable linear feature. The *Linear Representation Hypothesis* (Mikolov et al., 2013; Elhage et al., 2022; Park et al., 2024), stated for our case in Section 2, motivates the possibility of a *source-of-information* feature in a transformer's residual-stream activation space (Elhage et al., 2021): activations associated with reliance on retrieved evidence may differ systematically from those associated with parametric recall. This is a testable empirical hypothesis about how activation patterns vary with a model's relative reliance on retrieved context. It does not imply that each generated token has a single, uniquely identifiable source: retrieved evidence and parametric knowledge may jointly shape the same token.

We investigate this hypothesis by extracting several candidate **Context-Source Vectors** (CSVs) from contrastive prompt pairs, following representation-engineering and activation-steering methods (Zou et al., 2023; Turner et al., 2023). For each generated token, projecting its residual-stream activation onto a CSV yields a continuous, signed *context-reliance score*. The score measures alignment with the context-reliant pole of the contrast; it is not a categorical or causal attribution of where a token originated. Nevertheless, it provides an intrinsic, real-time proxy for the risk that a token is insufficiently grounded in retrieved context, without requiring additional model calls or external knowledge bases.

To compute these CSVs, we construct several synthetic contrastive prompt datasets using distinct templates and hallucination-prone scenarios. These include both direct extraction from context and inference or synthesis over contextual information. We derive a CSV from each construction for Llama-3.2-1B-Instruct and Llama-3.1-8B-Instruct, and evaluate their token-level predictive performance on the RAGTruth dataset (Niu et al., 2024). We compare them with randomly sampled activation-space directions and with an in-domain direction directly trained on RAGTruth, which serves as an empirical upper control on what the steering vector method can deliver. We additionally analyze the pairwise geometry of the derived CSVs to test whether context reliance corresponds to one stable shared direction or to multiple differently oriented but similarly predictive features.

The resulting score has important limitations. A low context-reliance score cannot distinguish accurate parametric knowledge from unsupported fabrication, and a high score does not by itself establish that a claim is entailed by the retrieved context. Moreover, multi-step reasoning or synthesis that genuinely uses contextual information may exhibit a different activation signature from direct contextual extraction, potentially reducing recall. Our direct-retrieval and inferential prompt constructions allow us to investigate this limitation empirically (Section 3). CSVs should therefore be understood not as replacements for factual verification, but as complementary, interpretable signals for monitoring, debugging, and understanding RAG behavior during inference.

Our experiments show that predictive context-reliance signals need not correspond to a unique shared activation-space direction. Across prompt constructions, some independently derived CSVs are closely aligned, whereas others are approximately orthogonal or negatively aligned while achieving similar token-level detection performance. We further find that CSV-based detection performs more strongly in Llama-3.1-8B-Instruct than in Llama-3.2-1B-Instruct. Together, these results suggest that context reliance is not captured by one stable linear feature; instead, multiple partly confounded activation-space features may be similarly predictive of downstream hallucination labels.

Contributions. (i) We test if context reliance is a single linearly decodable feature (Hypothesis 1). (ii) We release 15 contrastive datasets spanning direct extraction, multi-hop inference, decoy-in-context, and template-free dialog constructions, and the CSVs extracted from them for two model scales. (iii) We show that near-ceiling in-domain probe accuracy is a poor predictor of RAGTruth transfer, and that mutually orthogonal or negatively aligned directions transfer equally well — evidence that context-reliance signals are not carried by one canonical direction. (iv) We report matched random-direction and in-domain controls, which we recommend as the minimum evidential bar for steering-vector detectors.

2 Concepts and Related Work

Our method rests on the residual-stream view of transformers: information flows along a shared stream that attention heads and feed-forward blocks alternately read from and write to (Elhage et al., 2021).

Formally, at a token position t and after layer ℓ, the activation is $h_\ell^t \in \mathbb{R}^d$, where d is the embedding dimension.[1] The residual stream flows from h_0^t (post-embedding), to $h_{n_{\text{layers}}}^t$ (pre-unembedding), where n_{layers} is the total number of layers in the transformer. Each attention head and feed-forward block in between reads from these activations and writes back to them by adding their results on top.

For each layer ℓ, the vectors h_ℓ^t live in a d-dimensional space that encodes the information the transformer uses to compute the output tokens. The linear representation hypothesis (LRH) has been stated in different versions (Elhage et al., 2022; Park et al., 2024; Anonymous, 2026), but for this work, we stick to the following definition:

Hypothesis 1 *Sufficiently sophisticated transformer models contain the knowledge about whether specific tokens were primarily deduced from context information or purely from the model's internal memory (i.e. its parameters). This information is linearly represented in the token's activation h_ℓ^t.*

There are two claims here: (1) the model has this information, and (2) how to read it from the activation. Specifically, we assume that for some layer ℓ, there is a *steering vector* $v_\ell \in \mathbb{R}^d$ such that the inner product $\langle v_\ell, h_\ell^t \rangle$ is high for tokens derived from the context and low for tokens not derived from the context.

Such vectors for concepts other than "context-reliance" have been found to take a causal function: For example, modifying the inference of a model by adding a vector representing the Golden Gate Bridge mid-computation would make the model talk about the bridge in any context (Templeton et al., 2024).

This doesn't necessarily work for our more abstract concept. Indeed, O'Neill et al. (2025) report that adding a comparable probe-derived vector increases hallucination, but that *subtracting* it does so as well; the plausible reading is that perturbing the residual stream in either direction degrades the output rather than steering a specific concept. For a recent survey of evidence for and against the LRH across features, see Anonymous (2026).

Park et al. (2024) formalize the LRH using counterfactual pairs and connect it to both linear probing and steering interventions. This framing is particularly appropriate for our setting, since our datasets are constructed from paired prompts that differ in whether the answer should be drawn from the provided context or from parametric knowledge.

For RAG specifically, Sun et al. (2025) introduced ReDeEP, a white-box hallucination detection method. Instead of using steering vectors like us, they use *Logit Lens* (nostalgebraist, 2020) to predict the output token from intermediate activations and observe shifts in the output due to attention heads (which indicates context reliance) and feed-forward blocks (which they interpret as parametric knowledge).

Sun et al. (2025) additionally group existing RAG hallucination detectors into three families by how each treats the two knowledge sources. Methods that are *parametric confounded by external* read the hidden states alone, where the external context E is a common cause of both the parametric signal P they measure (Geva et al., 2021) and the hallucination H they predict. O'Neill et al. (2025) exemplify this family: they train a linear probe on an observer model's residual stream to expose and steer contextual hallucinations along a single direction, the closest prior work to ours. Methods that are *external confounded by parametric* read the context and the response, and are confounded in the reverse direction, since parametric knowledge is unavoidable in any response. Methods that *mix parametric and external*, typically through token probability or sampling, leave the two contributions entangled. In all three, the estimate of one source is contaminated by the other.

Unlike these approaches, we do not predict H directly. We instead estimate a direction separating the two knowledge stores themselves, with supervision coming from the construction of the contrastive pairs rather than from hallucination annotations, and read hallucination risk off as a projection onto that direction. The confounding structure above therefore never enters the estimation, though it does constrain what the resulting score can be claimed to measure (Section 3.2).

[1] d is typically between 1,024 (Llama 3.2 1B) and 16,384 (Llama 3.1 405B) in modern open-weight LLMs.

3 METHODOLOGY

3.1 COMPUTING THE CONTEXT-SOURCE STEERING VECTOR

We compute the Context-Source Vector (CSV) following *contrastive activation addition* (CAA) (Panickssery et al., 2024). Each pair of strings from the data sets (described in Section 3.2) gives two inputs: x^{ctx}, the prompt whose correct answer is the one the context supports, and x^{par}, the prompt whose correct answer is the one stored in the model's weights. The two have otherwise large overlap, so the source the answer must come from is what separates them. We run each through the frozen model and read the residual stream at one token position t^*: the appended answer letter for the A/B pairs, and the last token of the rewritten span for the RAGTruth pairs. The CSV at layer ℓ is the mean difference over the dataset D:

$$v_\ell := \frac{1}{|D|} \sum_{(x^{\text{ctx}},\, x^{\text{par}}) \in D} \left(h_\ell^{t^*}(x^{\text{ctx}}) - h_\ell^{t^*}(x^{\text{par}}) \right). \tag{1}$$

v_ℓ is modeling the difference in the internal representation between a context-reliant and a non-reliant token. We compute one vector per layer since we do not know in what layer this difference is represented, if at all. Averaging assumes the two classes differ along one shared direction, so that whatever varies from pair to pair cancels out and only the source distinction is left.

3.2 DATASET

A/B contrastive pairs. We build contrastive pairs from four sources. Most of our training data is synthetic, generated from 13 hand-written templates (details in Appendix A); to check that our conclusions are not tied to those templates, we also train 4200 prompts derived from ConFiQA (Bi et al., 2025), an external counterfactual QA corpus. ConFiQA is free-text, so we cast it into the same A/B format, using its counterfactual answer as the context answer and its original answer as the parametric one. In both cases, a pair consists of two forced-choice prompts offering the same two answers, labeled A and B, where the correct answer is the context-grounded one in the first prompt and the model's parametric (real-world) answer in the second. We draw one letter per pair at random and put the correct answer under that letter in both prompts, so the answer token stays fixed and only the source of the answer changes. We append that letter and run a single forward pass, reading the residual stream at its position, $h_\ell^{t^*}$, following CAA (Panickssery et al., 2024). The templates differ in how hard the context answer is to reach: in some it can be copied straight from the context, in others it takes reasoning hops, and in others the context holds both the context answer and the real one.

RAGTruth-derived pairs. The third source is RAGTruth (Niu et al., 2024), a corpus of real RAG responses with human-annotated hallucination spans, and it does not use the A/B format. For each annotated span, we ask an LLM for a minimal replacement that the context supports, keeping at most two spans per response. This gives two versions of the same prompt-and-response text that differ only in that span, and we read the residual stream at the span's last token. We use these pairs to build an in-domain reference vector: we fit it on the RAGTruth training split and score it (and all other vectors) on the RAGTruth test split. Because it is trained on the same kind of data it is tested on, it serves as an upper control, a best case that vectors trained on other data should not be expected to beat.

Generalizable dialog. The A/B format is convenient but unnatural, so our fourth source drops it entirely: we test whether a generic, concept-agnostic recipe — one that requires no hand-written template — recovers a comparable direction. First, we generate a dialog as starting point that contains three randomly selected words (see Appendix A for the prompts and details). Then we generate two possible continuations: one "neutral" continuation without further instructions, and one continuation where the model is asked to keep a certain concept in mind (for this paper specifically 'Repeating a fact from the first sentence').

Since without the A/B format, we do not have a specific token to measure, we take an average over all relevant tokens: for a fixed layer ℓ, the activations are extracted from all tokens T_n in the neutral

continuations, and all tokens T_p in the positive continuation. The steering vector corresponding to this contrastive pair is computed as the difference between the average activations:

$$v_\ell := \frac{1}{|T_p|} \sum_{t \in T_p} h_\ell^t - \frac{1}{|T_n|} \sum_{t \in T_n} h_\ell^t. \tag{2}$$

Many of the measured tokens will not represent the concept we are measuring, but these tokens are drawn from the same distribution as the neutral configuration, and are based on the same context, so over enough data points, the feature which is the only difference between the neutral and positive configuration will hopefully emerge from the averages. Possibly with a smaller magnitude than what a more targeted approach would achieve.

Context-source labeling. Our pairs label how an answer was computed, not how it reads: whether the answer token came from reading the context or from the model's weights. Faithfulness is a different thing, a relation between two texts, which is why post-hoc methods (NLI classifiers, SelfCheckGPT, LLM judges) can check it without ever running the model. Source cannot be checked that way: a token recalled from memory and the same token read from the context are identical as strings, so the difference exists only inside the computation. O'Neill et al. (2025) also read activations, but their label is still the string-level one; ours is internal at both ends. The two usually agree, and diverge when memorised recall happens to match the context, or when the model faithfully reads a wrong context. Our score is therefore a proxy for hallucination risk, not a factuality detector.

3.3 Scoring RAG Outputs

To test whether the directions trained on synthetic data transfer to hallucinations that occur naturally, we take the test split from our RAGTruth-derived dataset, and replay each prompt pair through the evaluated Llama model. For each last token of the prompts at position t we project its layer-ℓ activation onto the vector, $s_t = \langle v_\ell, h_\ell^t \rangle$. A high score means the token looks context-reliant, a low score means it looks more like recall from the model's weights. To avoid having an arbitrary choice of threshold influence the scoring, we compute the threshold-independent AUROC score. It gives the probability that for any randomly chosen pair of context-reliant and parametric prompts, the context-reliant one has the higher projection score.

Each CSV is computed on its own dataset's training split and then scored on the fixed RAGTruth test split, at every layer, always applying a layer-ℓ vector to layer-ℓ activations. It is never cross-validated, and it is trained and tested on different corpora. We call these the cross-dataset runs, and the AUROC they yield the *RAGTruth AUROC*.

3.4 Probe and Controls

Linear probe. Following O'Neill et al. (2025), we also fit an ℓ_2-regularised logistic-regression probe on the same answer-token activations used for the CSV, under nested five-fold cross-validation with an inner loop picking the layer, and we report the probe's mean F_1 and AUROC over the held-out outer folds. For `ragtruth-contrastive` the probe pools all three splits into that cross-validation. The probe score thus measures how separable a dataset's own pairs are, not how well its vector detects hallucinations on RAGTruth. It will tell us how well the essence of a dataset is represented in the model, and in what layer.

Two controls. The *upper* control is the CSV fitted on the RAGTruth training split (Section 3.2), scored on the same corpus it was trained on. The *lower* control is a random direction: 100 unit vectors drawn with no preferred direction, each held fixed across all layers and scored exactly as a CSV, so that its best layer is picked the same way as a template's.

4 Experimental Results

4.1 Linear Separability Across Layers

Before asking whether CSVs transfer, we check that the context-versus-parametric distinction is linearly present at all. We project the pooled answer-token activations $h_\ell^{t^*}$ onto their top two principal

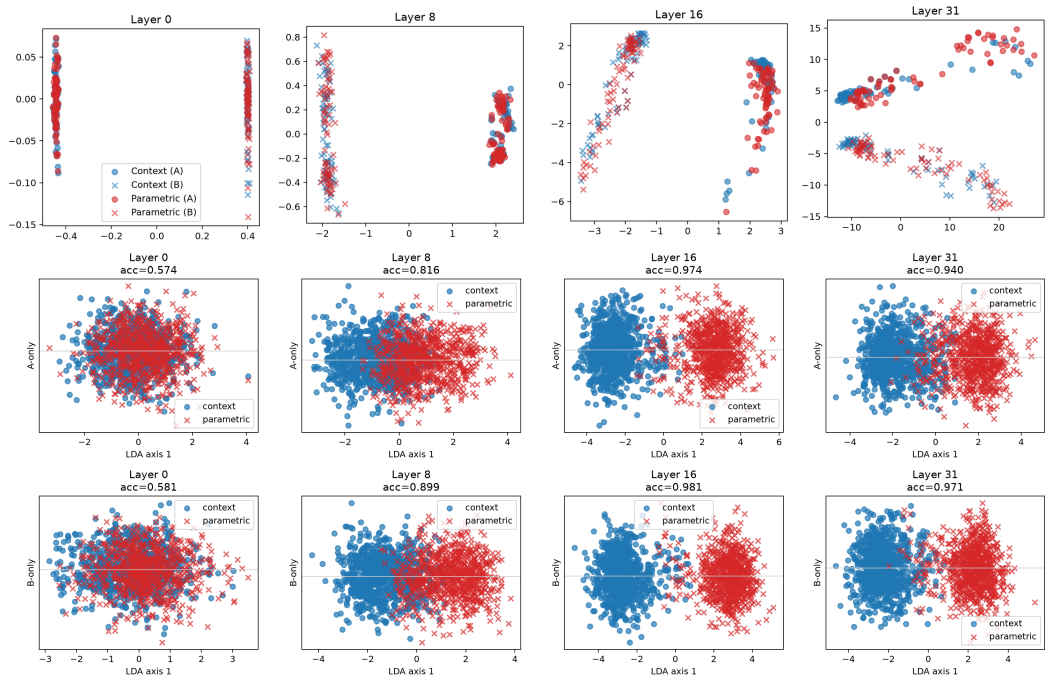


Figure 1: Answer-token activations for the pooled 11-template aggregate on Llama-3.1-8B-Instruct, at four representative decoder blocks of 32. Blue marks context-reliant and red parametric examples. **Top:** projection onto the top two principal components; circles denote pairs whose answer letter is A and crosses B. The components separate by answer letter at every depth, with the two classes overlapping inside each letter cluster. **Middle and bottom:** held-out test points projected onto the single discriminant of an LDA fit on the training split, restricted to A-only and B-only pairs respectively, with test accuracy above each panel. Holding the letter fixed removes it as a possible explanation for the split. Vertical spread in the lower two rows is jitter added for legibility and carries no information.

components, one panel per decoder block, for Llama-3.1-8B-Instruct and Llama-3.2-1B-Instruct. Vectors are computed at $n = 1000$ pairs per template (700/150/150 train/validation/test), with 2 exceptions, see Appendix B. Figure 1 shows four representative layers of the 8B model's 32; Figure 3 in Appendix C shows the same view for 1B. The dominant axis in every panel is the answer letter: pairs committing to A sit on one side and pairs committing to B on the other, with context and parametric points mixed inside each cluster. Answer-token identity is the highest-variance factor at that position, and it absorbs both components at every depth on both models.

A probe that holds the letter fixed does find the source distinction. On the 8B aggregate, held-out LDA accuracy rises from 0.574 (A-only, 1,634 test pairs) and 0.581 (B-only, 1,668) at layer 0 to 0.975 at layer 13 and 0.983 at layers 12–14, against a 0.5 baseline, and stays above 0.95 on both tasks from layer 10 to layer 30. The 4-class task reads 0.558 at layer 0 against a 0.25 baseline, which is what a probe recovering the letter and nothing else would score, and 0.973 at layer 14. The CSV itself peaks in the same band: AUROC 0.989 and accuracy 0.944 at layer 14, up from 0.539 at layer 0, with mean projections of $+1.29$ (context) against -0.93 (parametric) there and both below 10^{-3} in magnitude at layer 0. Llama-3.2-1B-Instruct peaks at layer 8 with a lower ceiling: 0.887 (A-only), 0.953 (B-only), 0.871 (4-class), and CSV AUROC 0.731, and unlike 8B, it falls back afterward (0.782 and 0.835 at layer 15). Because A-only and B-only separate independently, the split reflects the information source rather than which literal token the prompt ends in. This places the signal in the middle of the network, matching the depth at which Panickssery et al. (2024) report behavioral clustering and the layer range Section 4.2 identifies on Llama-3.1-8B-Instruct.

4.2 MIDDLE-LAYER SELECTION

We examine where the context-reliance signal is most clearly represented in Llama-3.1-8B-Instruct. For each prompt family and layer, we project held-out final-token residual-stream activations onto the corresponding steering vector, estimated from the training split; higher projections indicate context-reliant examples and lower projections indicate parametric examples.

Separation is weak or inconsistent in the earliest layers (8–12) and becomes clear from around layer 14 onward, though not uniformly across templates: most separate cleanly by layer 14, while a subset remain overlapped until layer 20–21 (Appendix E breaks this down by template and construction). Layers 14–18 provide the most consistent distinction on average, and layer 16, the midpoint of the model's 32 transformer blocks, lies within this stable region. Later layers show larger absolute dot-product differences, but also substantially wider and more overlapping distributions, so the larger raw gap does not translate into a cleaner class boundary. We therefore use layer 16 as representative: it provides strong average separation without the increased output-layer variance, but is not claimed optimal for every prompt or architecture.

4.3 DOES PROBE ACCURACY PREDICT RAGTRUTH AUROC?

Near-perfect in-domain probe accuracy is tempting to read as evidence that the model represents the distinction we intended. This section asks whether that evidence is diagnostic: if a dataset's two classes are almost perfectly decodable, does the direction it yields detect *naturally occurring* hallucinations?

We compare the probe with the RAGTruth AUROC of each dataset's vector, against the two controls of Section 3.4.

Random baseline. The random directions centre on 0.5, but scatter four to six times more widely than they would if the spans were independent, since the two members of a pair differ in one span only. At 1B the 95th percentile of their Best is 0.634, and at the middle layer 0.586 (Table 1).

Results. Full LR and AUROC scores can be found in Table 1. At 1B, the upper control reaches 0.770 at layer 8, far away from the 1.0 that a perfect detection would yield, showing the limits of the steering vector approach in this model. The 8B model achieves 0.841 for the upper control. Prompts achieve varied results: For 1B, the best vectors are `general_dialog` (0.681), `prompt_9` (0.676), `prompt_yes_no` (0.667), `prompt_11` (0.656) and `prompt_1` (0.641), clearing the random 95th percentile of 0.634 on different layers. The worst datasets lie around the average for a random vector. At the middle layer specifically, only `prompt_4` (0.619) and `general_dialog` (0.609) clear the random-direction 95th percentile of 0.586. Ten of the 15 templates peak at layers 0–3 and fall back towards 0.5 at the middle layer, so their Best reflects the choice of layer more than a stable signal. For 8B, the numbers for the best layer are better across the board, but the middle layer is a bad choice for quite a number of prompts. `prompt_yes_no` is especially interesting, since the 1B model wasn't even able to separate out the test data in the same format as the training data (F_1 is 0.753), while for 8B, this prompt has the fourth highest scores.

The selected layer says more. Layer selection takes the argmax of inner-fold F_1 and breaks ties to the lowest index, so on a dataset at ceiling ℓ_p is the *earliest layer at ceiling*, not a located optimum. Read that way it tracks how shallow each dataset's shortcut is. `prompt_3` and `prompt_4` reach $F_1 = 1.000$ by layers 2–3 on both models and use invented entities ("Valdoria"); a feature decodable that early is most plausibly lexical, a clue absent from RAGTruth, whose hallucinations are fluent and use real vocabulary. Most other templates settle in the middle of the network, at layers 8–10 for 1B and 10–16 for 8B, close to the upper control's 7 and 14; `confiqa` (8 and 14) is consistent with its context answer appearing verbatim in the passage. Only `prompt_10`, `prompt_11` and `prompt_semantic` stay clearly below ceiling on both models (F_1 0.84–0.95), so only for them is ℓ_p a genuine optimum. `prompt_10` and `prompt_11` place both answers in the passage, and at 8B their RAGTruth AUROC peaks late, at layers 22 and 23, consistent with later layers handling entity–attribute relations (Geva et al., 2021). More generally, the probe's layer is a poor guide to where the vector works on RAGTruth: at 8B, ℓ_p and ℓ_v differ by more than five layers for 11 of the 15 templates.

Table 1: Probe F_1 and RAGTruth AUROC for the CSVs. ℓ_p: layer selected by the probe. Best: highest RAGTruth AUROC over all layers, reached at layer ℓ_v. Mid: RAGTruth AUROC at a fixed middle layer (8 for 1B, 16 for 8B). Each random row is one random direction, fixed across layers and scored the same way; the last two rows summarise 100 of them. `all` pools the 11 generated templates.

	1B (16 layers)					**8B** (32 layers)				
Dataset	F_1	ℓ_p	Best	ℓ_v	Mid	F_1	ℓ_p	Best	ℓ_v	Mid
Upper control — no domain gap										
`ragtruth-contrastive`	0.821	7	*0.770*	8	0.770	0.854	14	*0.841*	13	0.827
Cross-dataset — fitted elsewhere, scored on RAGTruth										
`confiqa`	0.953	8	0.596	10	0.529	0.998	14	0.699	7	0.522
`all`	0.952	8	0.626	2	0.554	0.971	13	0.666	23	0.608
`prompt_1`	1.000	5	0.641	2	0.520	0.999	7	0.636	28	0.509
`prompt_2`	0.999	7	0.631	2	0.527	0.999	14	0.637	3	0.408
`prompt_3`	1.000	2	0.589	3	0.541	1.000	2	0.652	14	0.594
`prompt_4`	1.000	2	0.619	8	0.619	1.000	3	0.655	23	0.629
`prompt_7`	1.000	8	0.616	2	0.581	1.000	10	0.633	9	0.439
`prompt_8`	0.994	9	0.589	10	0.532	1.000	10	0.693	12	0.622
`prompt_9`	1.000	8	0.676	7	0.516	0.999	13	0.591	9	0.408
`prompt_10`	0.897	8	0.626	1	0.384	0.913	12	0.651	22	0.608
`prompt_11`	0.839	9	0.656	1	0.416	0.860	15	0.648	23	0.620
`prompt_ambiguous`	0.975	8	0.561	1	0.432	1.000	10	0.645	23	0.595
`prompt_fakefacts`	0.976	8	0.566	0	0.530	1.000	16	0.632	23	0.595
`prompt_semantic`	0.914	10	0.583	2	0.445	0.952	13	0.627	11	0.586
`prompt_yes_no`	0.753	5	0.667	15	0.563	0.980	21	0.669	13	0.614
`general_dialog`			0.681	15	0.609			0.693	10	0.553
Lower control — random direction, fixed across layers										
Random 1			0.540	14	0.465			0.587	14	0.546
Random 2			0.536	0	0.493			0.618	12	0.542
Random 3			0.444	4	0.368			0.565	26	0.521
Random 4			0.560	2	0.466			0.526	26	0.473
Random 5			0.553	8	0.553			0.542	13	0.470
Random, mean (n=100)			*0.572*		*0.495*			*0.580*		*0.496*
Random, 95th pct.			*0.634*		*0.586*			*0.645*		*0.601*

4.4 Geometry of the extracted directions

If context reliance were a single linear feature, all CSVs would be tightly clustered. Figure 2 shows that they are not. As should be expected, the random vectors are almost orthogonal to each other and to our prompts. The vector 'general_dialog' generated from dialogs instead of multiple choice is also remarkably orthogonal to all other prompts, stemming most likely from the fact that it is the only vector not generated from only "A" or "B" tokens, but an average of a whole phrase instead. The other prompts are closer in angle, but spread far enough apart for some pairs to still make obtuse angles (blue lines).

In the 1B model, prompt_yes_no stands out as being negatively correlated with all other vectors, while being in the middle of a close cluster in the 8B model. This together with the low F_1 score on its own data set for the 1B model suggests that the captured concept is just barely represented inside the small model, but present and correlated with concept reliance in the large model.

In the 8B model, prompt_7 and prompt_9 are strongly aligned to each other and remarkably far apart from all other vectors, almost antipodal. This correlates with their low AUROC score of 0.439 and 0.408 on that specific layer 16 (out of 32), while their best AUROC scores can be found at layer 9 (0.633 and 0.591). This suggests very visually that whatever concept these prompts represent, it is present in layer 9, while the other prompts capture concepts that are at play in different layers of the model.

Since most of these vectors negatively correlate with hallucinations, they do need to capture different aspects of context reliance, thereby providing evidence against the simplest interpretation of

Hypothesis 1. Context reliance likely has different components to it that can be measured in different layers of the model.

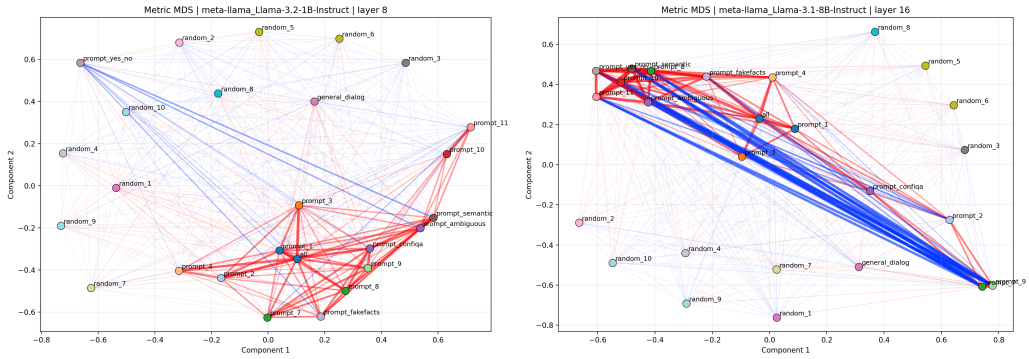


Figure 2: Visualizing the closeness of our computed steering vectors via metric multidimensional scaling, using $1-$ cosine similarity as metric. Red lines indicate cosine similarity > 0, blue lines < 0, thickness indicates magnitude. See Appendix D for numerical values. **Left:** Llama-3.2-1B, layer 8 of 16 **Right:** Llama-3.1-8B, layer 16 of 32.

5 Conclusions and Future Work

This work investigated whether residual-stream activations contain a signal of a model's reliance on retrieved context that can indicate hallucination risk in RAG. We derived Context-Source Vectors (CSVs) from contrastive prompt pairs and evaluated their token-level transfer to RAGTruth. The context-versus-parametric distinction is linearly separable in synthetic pairs, especially in middle layers, and is represented more strongly and persistently in Llama-3.1-8B-Instruct than in Llama-3.2-1B-Instruct.

Transfer to naturally occurring RAG hallucinations is promising but limited. Several cross-dataset vectors exceed random directions, reaching best-layer AUROCs of up to 0.681 for 1B and 0.699 for 8B, while RAGTruth-fitted vectors reach 0.770 and 0.841. Performance varies substantially across prompt constructions and layers, and best transfer layers often differ from middle and probe-selected layers. Thus, the results support source-related activation signals, but not yet a robust, deployment-ready detector based on a single steering vector.

The geometric analysis further challenges the notion of one prompt-invariant context-reliance direction. Some independently derived CSVs are closely aligned, whereas others are approximately orthogonal or negatively aligned despite achieving similar downstream performance. This suggests that the extracted directions may capture different mixtures of context reliance, semantic relations, answer format, lexical cues, and other task-specific shortcuts. Accordingly, a CSV projection should be interpreted as a continuous proxy for context reliance rather than a causal attribution of a token's origin or a test of factual entailment: a low score cannot distinguish correct parametric recall from unsupported fabrication, and a high score cannot guarantee that a claim is supported by a correct retrieved context.

Future work should develop more naturalistic contrastive datasets with stronger controls for lexical, formatting, and template-specific confounds, and evaluate the approach across additional models, scales, languages, retrieval conditions, and RAG tasks. Rather than selecting a single vector and layer, future methods could learn and validate a low-dimensional context-reliance subspace, combine complementary directions across layers, and calibrate token-level scores into span- or response-level warnings. Combining these inexpensive white-box signals with external entailment verification, uncertainty estimation, or retrieval-quality measures may yield more reliable hallucination detection. Finally, causal interventions and component-level analyses are needed to determine whether the measured directions truly track the use of retrieved evidence rather than correlated properties of the prompts or generated text.

AI USE STATEMENT

Generative AI tools assisted with literature search, coding, dataset creation (as described in the paper), drafting of individual paragraphs, language editing and polishing. The authors are responsible for the final content, claims, and artifacts in this work.

REPRODUCIBILITY STATEMENT

Appendices A and B specify the full data construction, i.e., the 13 prompt templates, the generation and filtering instructions, the ConFiQA and RAGTruth derivations, and all splits. All prompts, generated datasets, extracted steering vectors, and evaluation scripts are included in the anonymized Supplementary Material, with a README that reproduces every number in Table 1 from the released vectors.

REFERENCES

Anonymous. A survey on the linear representation hypothesis. In *Submitted to ACL Rolling Review - March 2026*, 2026. URL https://openreview.net/forum?id=EUf81VIG23. under review.

Baolong Bi, Shaohan Huang, Yiwei Wang, Tianchi Yang, Zihan Zhang, Haizhen Huang, Lingrui Mei, Junfeng Fang, Zehao Li, Furu Wei, Weiwei Deng, Feng Sun, Qi Zhang, and Shenghua Liu. Context-DPO: Aligning language models for context-faithfulness. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 10280–10300, Vienna, Austria, July 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-acl.536. URL https://aclanthology.org/2025.findings-acl.536/.

Souvick Das, Sallam Abualhaija, and Domenico Bianculli. How much do legal rag systems still hallucinate? *arXiv:2608.14210*, August 2026. doi: 10.48550/arXiv.2608.14210. URL http://arxiv.org/abs/2608.14210. arXiv:2608.14210 [cs.CL].

Nelson Elhage, Neel Nanda, Catherine Olsson, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. *Transformer Circuits Thread*, 2022. https://transformer-circuits.pub/2022/toy_model/index.html.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. A survey on RAG meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024, Barcelona, Spain, August 25-29, 2024*, pp. 6491–6501. ACM, 2024. doi: 10.1145/3637528.3671470.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *EMNLP*, 2021.

Or Honovich, Leshem Choshen, Roee Aharoni, Ella Neeman, Idan Szpektor, and Omri Abend. TRUE: Re-evaluating factual consistency evaluation. In *NAACL*, 2022.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Computing Surveys*, 2025.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 2023.

Saurav Kadavath, Tom Conerly, Amanda Askell, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *NeurIPS*, 2020.

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

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. Entity-based knowledge conflicts in question answering. In *EMNLP*, 2021.

Potsawee Manakul, Adian Liusie, and Mark J.F. Gales. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *EMNLP*, 2023.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*, 2024.

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

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In Lucy Vanderwende, Hal Daumé III, and Katrin Kirchhoff (eds.), *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 746–751, Atlanta, Georgia, June 2013. Association for Computational Linguistics. URL https://aclanthology.org/N13-1090/.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10862–10878, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.585. URL https://aclanthology.org/2024.acl-long.585/.

nostalgebraist. Interpreting gpt: the logit lens. *AI Alignment Forum*, 2020. URL https://www.alignmentforum.org/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens.

Charles O'Neill, Slava Chalnev, Chi Chi Zhao, Max Kirkby, and Mudith Jayasekara. A single direction of truth: An observer model's linear residual probe exposes and steers contextual hallucinations. *arXiv preprint arXiv:2507.23221*, 2025.

Nina Panickssery, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering llama 2 via contrastive activation addition. *arXiv preprint arXiv:2312.06681*, 2024.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 39643–39666. PMLR, 2024. URL https://proceedings.mlr.press/v235/park24c.html.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. Retrieval augmentation reduces hallucination in conversation. In *Findings of EMNLP*, 2021.

Zhongxiang Sun, Xiaoxue Zang, Kai Zheng, Yang Song, Jun Xu, Xiao Zhang, Weijie Yu, and Han Li. ReDeEP: Detecting hallucination in retrieval-augmented generation via mechanistic interpretability. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2410.11414.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L. Turner, Callum McDougall, Monte MacDiarmid, Alex Tamkin, Esin Durmus, Tristan Hume, Francesco Mosconi, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, Shan Carter, Chris Olah, and Tom Henighan. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits*, May 2024. URL https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Activation addition: Steering language models without optimization. *arXiv preprint arXiv:2308.10248*, 2023.

Benjamin Weiser. Here's what happens when your lawyer uses ChatGPT. *The New York Times*, 2023.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*, 2023.

A Details of Synthetic Data Generation

The A/B templates, ConFiQA and RAGTruth pairs are described in Appendix B; this section covers the dialog pairs behind `general_dialog`.

For each of 300 items we drew three random words and asked: `Please generate a short dialog of at most 5 sentences, containing the words '{w1}', '{w2}', '{w3}'. Respond with nothing else.` We then asked twice for a continuation, `Please add exactly one sentence to this dialog. Output only this sentence, nothing else.`, once as is (neutral) and once followed by `While doing so, keep the concept 'Repeating a fact from the first sentence' in mind.` (positive). The vector is computed over the continuation tokens only, as in Section 3.2. The items are split 210/45/45.

B Details of Data Construction

We construct contrastive pairs containing a *context-reliant* example, whose target follows the supplied context, and a *parametric* example, whose target follows world knowledge or the designated non-contextual answer. Templates 1–4 ask different questions over a shared statement; Templates 5–11, prompt_ambiguous, prompt_yes_no and ConFiQA keep the question fixed and reverse the answer mapping. Templates 10–11 additionally place the parametric answer in the passage as a decoy. RAGTruth pairs contrast an annotated hallucinated span with a minimal source-supported correction.

B.1 Generation and Pair Construction

Templates 1–11 were generated as structured JSON tables (1000 rows per template). The generator was instructed to return only the requested fields, use concise, non-duplicated English values, and avoid sensitive content. Malformed outputs, missing fields, and invalid answer labels were rejected. When talking about 11 pooled templates, these are the ones that were pooled. Templates prompt_ambiguous and prompt_yes_no are fixed, manually curated datasets of 100 examples each. The common generation instruction was:

```
Generate exactly {num_rows} JSON records for Template
{template_id}: {template_name}, using only {required_columns}
and following {template_specific_constraints}. Keep values
```

Table 2: Prompt templates. CR and P denote context-reliant and parametric members.

Template	Prompt text
prompt_1	CR: `The {entity_type} of {fictional_place} is {fictional_fact}. What is the {entity_type} of {fictional_place}?` P: same context; ask for {fact_type} of {real_place}.
prompt_2	CR: assign {fictional_alias} to {real_entity_1} and query that entity. P: keep the alias statement but query {real_entity_2} under {contextual_condition_2}.
prompt_3	CR: define {fictional_term} in {fictional_domain} and ask for its definition. P: keep the definition but ask for {fact_type} of {real_place}.
prompt_4	CR: state an explicit alias rule for {real_entity_1} and query it. P: keep the rule but query {real_entity_2}.
prompt_fakefacts	`According to an ancient myth, {context}. {question}?` Targets: contextual/real answer.
prompt_semantic	`Context: {context}. Question: {question}?` Targets: contextual/real answer.
prompt_7	A biography transcript claims that {real_person} invented {fictional_invention}; query what the person invented. Targets: fictional/real invention.
prompt_8	A JSON log assigns {fictional_classification} to {real_entity}; query the classification. Targets: fictional/real classification.
prompt_9	A cloze passage calls {fictional_capital} the capital of {real_country}. Targets: fictional/real capital.
prompt_10	Assert {fictional_answer} for a target attribute; mention {real_answer} as an attribute of a decoy entity.
prompt_11	Assert {fictional_answer} for a target attribute; include {real_answer} only in an incidental label or note.
prompt_ambiguous	Short contextual statement plus an underspecified open question. Targets: contextually salient/plausible alternative answer.
prompt_yes_no	Short factual statements plus a yes/no question. Targets: context-supported/opposite-polarity answer.

```
concise, avoid duplicates, use only English, avoid sensitive
content, and return only the JSON array.
```

All multiple-choice prompts used the system message `You are a helpful assistant.` `Answer with only the letter A or B.` and ended with `Provide only the letter of your chosen answer (A or B).` Answer positions were balanced. Within each pair, the same target letter was associated with different answer text, preventing the contrast from being explained by the final answer label. Data were split 70/15/15 into training, validation, and test sets.

B.2 Prompt Templates

Table 2 summarises all prompt families. Braces denote row-specific fields; every prompt is followed by options A/B and the shared letter-only instruction above. The inconsistent naming convention and missing prompt_5 and prompt_6 is due to discarding and merging prompt lists created by different people and was kept to avoid mislabeling accidents.

B.3 Externally Derived Data

ConFiQA. We use the QA subset. The counterfactual context and answer become the passage and context-reliant target; the original answer becomes the parametric target. Records with missing or identical answers are removed and answer positions are balanced. This produces 5,999 examples using the Template 6 prompt, which are split 4200/899/900 into training, validation and test.

RAGTruth. We retain spans labelled *Evident Conflict* or *Evident Baseless Info*, using at most two spans per response. A language model is instructed to return only a minimal, source-supported replacement of similar length and style. Empty, unchanged, or non-local rewrites are discarded. The resulting dataset contains 6,673/1,669/1,166 train/validation/test pairs.

Table 3: Representative contrastive pairs across the dataset constructions described in Section 3.2.

Construction	Context-reliant prompt	Parametric prompt
Two different questions (Template 1)	Context: The capital of Zoria is Brindlemark. Question: What is the capital of Zoria? A) Brindlemark B) Paris *Correct: A*	Context: The capital of Zoria is Brindlemark. Question: What is the capital of France? A) Brindlemark B) Paris *Correct: B*
Two different questions (Template 3)[2]	Definition: In competitive falconry, the term "a hollow strike" means a scoring maneuver where the falcon touches the lure without gripping it. Question: According to this definition, what does "a hollow strike" mean? A) a scoring maneuver where the falcon touches the lure without gripping it B) Tokyo *Correct: A*	Definition: In competitive falconry, the term "a hollow strike" means a scoring maneuver where the falcon touches the lure without gripping it. Question: Outside of competitive falconry, what is the capital of Japan? A) a scoring maneuver where the falcon touches the lure without gripping it B) Tokyo *Correct: B*
Forced conflict, no decoy (Semantic / ConFiQA)	Context: According to the archive, the capital of France is Berlin. Question: What is the capital of France? A) Berlin B) Paris *Correct: A*	Context: According to the archive, the capital of France is Berlin. Question: What is the capital of France? A) Berlin B) Paris *Correct: B*
Forced conflict, with decoy (Template 10)	Context: The archive summary lists the population of Rivemoor as 40,000. A separate note about Kessendale gives its population as 62,000, noting both towns share the same river valley. Question: What is the population of Rivemoor? A) 40,000 B) 62,000 *Correct: A*	Same passage as at left. Question: What is the population of Rivemoor? A) 40,000 B) 62,000 *Correct: B*
Forced conflict, incidental decoy (Template 11)	Field Notes: The report claims that the completion year of the Sydney Opera House is 1968. Nearby, a commemorative fountain is mentioned only because its dedication plaque includes the words "1973." Visitors often pause there before touring the harbour. Question: What is the completion year of the Sydney Opera House? A) 1968 B) 1973 *Correct: A*	Same passage as at left. Question: What is the completion year of the Sydney Opera House? A) 1968 B) 1973 *Correct: B*

B.4 Dataset Examples

Table 3 shows representative examples from five of the thirteen templates. Templates 1 and 3 ask different questions over a shared fact, while Semantic/ConFiQA and Templates 10–11 pose a forced conflict between the context-supported and parametric answers, without and with an embedded decoy, respectively.

Table 4: RAGTruth-derived contrastive pairs, built from annotated hallucination spans of both dominant label types.

Field	Text
Source (excerpt)	"...new research released by the Anne Frank House shows that Anne and her older sister, Margot Frank, died at least a month earlier than previously thought... probably did not survive to March 1945..."
Hallucinated (Evident Conflict)	"...Anne and Margot likely died... believed to have died before February 7, 2022..."
Corrected	"...Anne and Margot likely died... believed to have died before February 7, 1945..."
Hallucinated (Evident Baseless Info)	"...at least a month earlier than previously believed, has prompted the Anne Frank House to issue a corrected statement regarding the date of her passing..."
Corrected	"...at least a month earlier than previously believed, a conclusion based on new research released by the Anne Frank House regarding the date of her passing..."

Table 4 shows two naturalistic RAGTruth-derived pairs covering both dominant label categories. Both responses were generated by `mistral-7B-instruct` for the same CNN/DailyMail summarisation query. Underlined spans mark the human-annotated labels and their minimally rewritten replacements.

[2]In an early version, template 3's context-reliant answers systematically began with an indefinite article ("a scoring maneuver..."), a known surface confound flagged during dataset auditing; after noticing we normalized the articles of all templates.

C PCA AND LDA ON LLAMA-3.2-1B-INSTRUCT

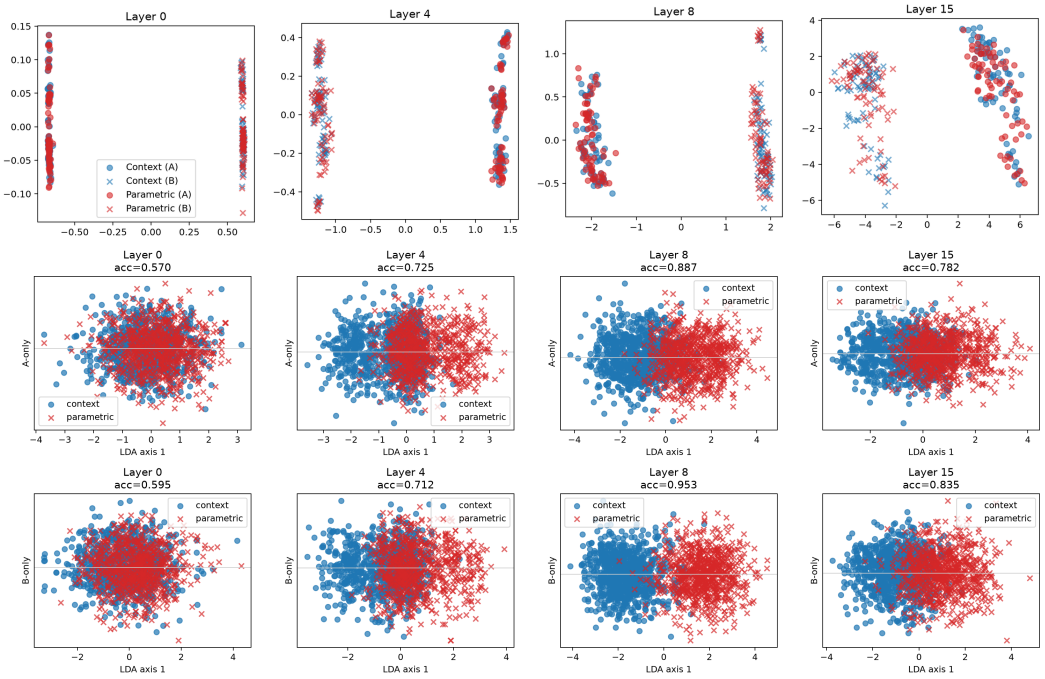


Figure 3: The view of Figure 1 for Llama-3.2-1B-Instruct, at four decoder blocks of 16. The letter again dominates the top two principal components at every depth. Letter-controlled LDA accuracy peaks at layer 8 (0.887 A-only, 0.953 B-only) and drops to 0.782 and 0.835 by the last layer, where the 8B model stays at 0.94 or higher.

D PAIRWISE COSINE SIMILARITIES

See tables 5 and 6 for the numerical pairwise cosine similarities and magnitudes of the computed steering vectors. The random vectors were left out, their cosine similarities with everything was mostly between -0.02 and +0.02, with very few outliers of up to +-0.058.

Table 5: Pairwise cosine similarities — model: meta-llama/Llama-3.2-1B-Instruct — layer: 8

Layer 8	all	prompt_1	prompt_2	prompt_3	prompt_4	prompt_7	prompt_8	prompt_9	prompt_10	prompt_11	prompt_ambiguous	prompt_confiqa	prompt_fakefacts	prompt_semantic	prompt_yes_no	general_dialog	magnitude
all	+1.000	+0.824	+0.711	+0.712	+0.587	+0.501	+0.582	+0.544	+0.196	+0.009	+0.461	+0.602	+0.521	+0.420	-0.166	+0.062	0.2901
prompt_1	+0.824	+1.000	+0.536	+0.495	+0.338	+0.341	+0.429	+0.390	+0.064	-0.048	+0.368	+0.442	+0.344	+0.333	-0.177	+0.057	0.9206
prompt_2	+0.711	+0.536	+1.000	+0.433	+0.503	+0.313	+0.324	+0.267	+0.040	-0.117	+0.108	+0.248	+0.262	+0.048	+0.031	-0.012	0.5391
prompt_3	+0.712	+0.495	+0.433	+1.000	+0.291	+0.182	+0.278	+0.265	+0.143	+0.091	+0.312	+0.305	+0.223	+0.295	-0.134	+0.046	0.8840
prompt_4	+0.587	+0.338	+0.503	+0.291	+1.000	+0.108	+0.152	+0.171	-0.011	-0.115	+0.036	+0.200	+0.205	+0.036	-0.093	+0.036	0.8359
prompt_7	+0.501	+0.341	+0.313	+0.182	+0.108	+1.000	+0.483	+0.532	+0.031	-0.236	+0.258	+0.502	+0.437	+0.059	+0.060	+0.070	0.3804
prompt_8	+0.582	+0.429	+0.324	+0.278	+0.152	+0.483	+1.000	+0.446	+0.067	-0.077	+0.459	+0.587	+0.465	+0.430	-0.105	+0.041	0.2807
prompt_9	+0.544	+0.390	+0.267	+0.265	+0.171	+0.532	+0.446	+1.000	+0.234	-0.014	+0.324	+0.624	+0.203	+0.236	+0.078	+0.098	0.2921
prompt_10	+0.196	+0.064	+0.040	+0.143	-0.011	+0.031	+0.067	+0.234	+1.000	+0.710	+0.329	+0.229	+0.071	+0.293	+0.042	-0.025	0.1537
prompt_11	+0.009	-0.048	-0.117	+0.091	-0.115	-0.236	-0.077	-0.014	+0.710	+1.000	+0.310	+0.074	-0.082	+0.368	-0.045	-0.051	0.1676
prompt_ambiguous	+0.461	+0.368	+0.108	+0.312	+0.036	+0.258	+0.459	+0.324	+0.329	+0.310	+1.000	+0.557	+0.396	+0.791	-0.336	+0.066	0.3157
prompt_confiqa	+0.602	+0.442	+0.248	+0.305	+0.200	+0.502	+0.587	+0.624	+0.229	+0.074	+0.557	+1.000	+0.482	+0.540	-0.155	+0.117	0.1892
prompt_fakefacts	+0.521	+0.344	+0.262	+0.223	+0.205	+0.437	+0.465	+0.203	+0.071	-0.082	+0.396	+0.482	+1.000	+0.368	-0.253	+0.030	0.3960
prompt_semantic	+0.420	+0.333	+0.048	+0.295	+0.036	+0.059	+0.430	+0.236	+0.293	+0.368	+0.791	+0.540	+0.368	+1.000	-0.369	+0.015	0.3278
prompt_yes_no	-0.166	-0.177	+0.031	-0.134	-0.093	+0.060	-0.105	+0.078	+0.042	-0.045	-0.336	-0.155	-0.253	-0.369	+1.000	+0.003	0.2153
general_dialog	+0.062	+0.057	-0.012	+0.046	+0.036	+0.070	+0.041	+0.098	-0.025	-0.051	+0.066	+0.117	+0.030	+0.015	+0.003	+1.000	0.9070
Scale	-1.000	-0.667	-0.333	+0.000	+0.333	+0.667	+1.000										

Table 6: Pairwise cosine similarities — model: meta-llama/Llama-3.1-8B-Instruct — layer: 16

Layer 16	all	prompt_1	prompt_2	prompt_3	prompt_4	prompt_7	prompt_8	prompt_9	prompt_10	prompt_11	prompt_ambiguous	prompt_confiqa	prompt_fakefacts	prompt_semantic	prompt_yes_no	general_dialog	magnitude
all	+1.000	+0.687	+0.424	+0.558	+0.684	+0.070	+0.548	+0.022	+0.447	+0.315	+0.499	+0.607	+0.715	+0.401	+0.202	+0.047	1.2272
prompt_1	+0.687	+1.000	+0.297	+0.507	+0.439	-0.037	+0.298	-0.025	+0.227	+0.153	+0.378	+0.370	+0.387	+0.265	+0.092	+0.012	2.2096
prompt_2	+0.424	+0.297	+1.000	+0.053	+0.293	+0.411	-0.185	+0.470	-0.261	-0.343	-0.201	+0.272	+0.089	-0.308	-0.361	+0.085	2.2554
prompt_3	+0.558	+0.507	+0.053	+1.000	+0.312	-0.070	+0.278	-0.084	+0.211	+0.220	+0.377	+0.362	+0.280	+0.270	+0.124	+0.021	2.6012
prompt_4	+0.684	+0.439	+0.293	+0.312	+1.000	-0.101	+0.427	-0.181	+0.287	+0.179	+0.291	+0.224	+0.583	+0.335	+0.239	-0.002	2.9014
prompt_7	+0.070	-0.037	+0.411	-0.070	-0.101	+1.000	-0.661	+0.876	-0.678	-0.659	-0.500	+0.502	-0.330	-0.776	-0.792	+0.095	5.1590
prompt_8	+0.548	+0.298	-0.185	+0.278	+0.427	-0.661	+1.000	-0.706	+0.819	+0.713	+0.707	+0.060	+0.719	+0.882	+0.777	-0.042	4.0572
prompt_9	+0.022	-0.025	+0.470	-0.084	-0.181	+0.876	-0.706	+1.000	-0.694	-0.655	-0.542	+0.395	-0.396	-0.823	-0.784	+0.091	4.4737
prompt_10	+0.447	+0.227	-0.261	+0.211	+0.287	-0.678	+0.819	-0.694	+1.000	+0.872	+0.694	-0.014	+0.604	+0.854	+0.802	-0.048	2.7695
prompt_11	+0.315	+0.153	-0.343	+0.220	+0.179	-0.659	+0.713	-0.655	+0.872	+1.000	+0.670	-0.092	+0.425	+0.725	+0.743	-0.045	1.8168
prompt_ambiguous	+0.499	+0.378	-0.201	+0.377	+0.291	-0.500	+0.707	-0.542	+0.694	+0.670	+1.000	+0.147	+0.597	+0.764	+0.571	+0.015	2.0803
prompt_confiqa	+0.607	+0.370	+0.272	+0.362	+0.224	+0.502	+0.060	+0.395	-0.014	-0.092	+0.147	+1.000	+0.274	-0.053	-0.285	+0.070	1.2809
prompt_fakefacts	+0.715	+0.387	+0.089	+0.280	+0.583	-0.330	+0.719	-0.396	+0.604	+0.425	+0.597	+0.274	+1.000	+0.651	+0.480	-0.006	2.5921
prompt_semantic	+0.401	+0.265	-0.308	+0.270	+0.335	-0.776	+0.882	-0.823	+0.854	+0.725	+0.764	-0.053	+0.651	+1.000	+0.814	-0.060	4.1949
prompt_yes_no	+0.202	+0.092	-0.361	+0.124	+0.239	-0.792	+0.777	-0.784	+0.802	+0.743	+0.571	-0.285	+0.480	+0.814	+1.000	-0.079	3.8057
general_dialog	+0.047	+0.012	+0.085	+0.021	-0.002	+0.095	-0.042	+0.091	-0.048	-0.045	+0.015	+0.070	-0.006	-0.060	-0.079	+1.000	1.9475
Scale	-1.000	-0.667	-0.333	+0.000	+0.333	+0.667	+1.000										

E LAYER SELECTION DIAGNOSTICS

Figure 4 shows the held-out validation distributions underlying the layer selection in Section 4.2: for each of the 13 templates and each layer of Llama-3.1-8B-Instruct, the projection of context-reliant (green) and parametric (orange) validation examples onto that template's steering vector.

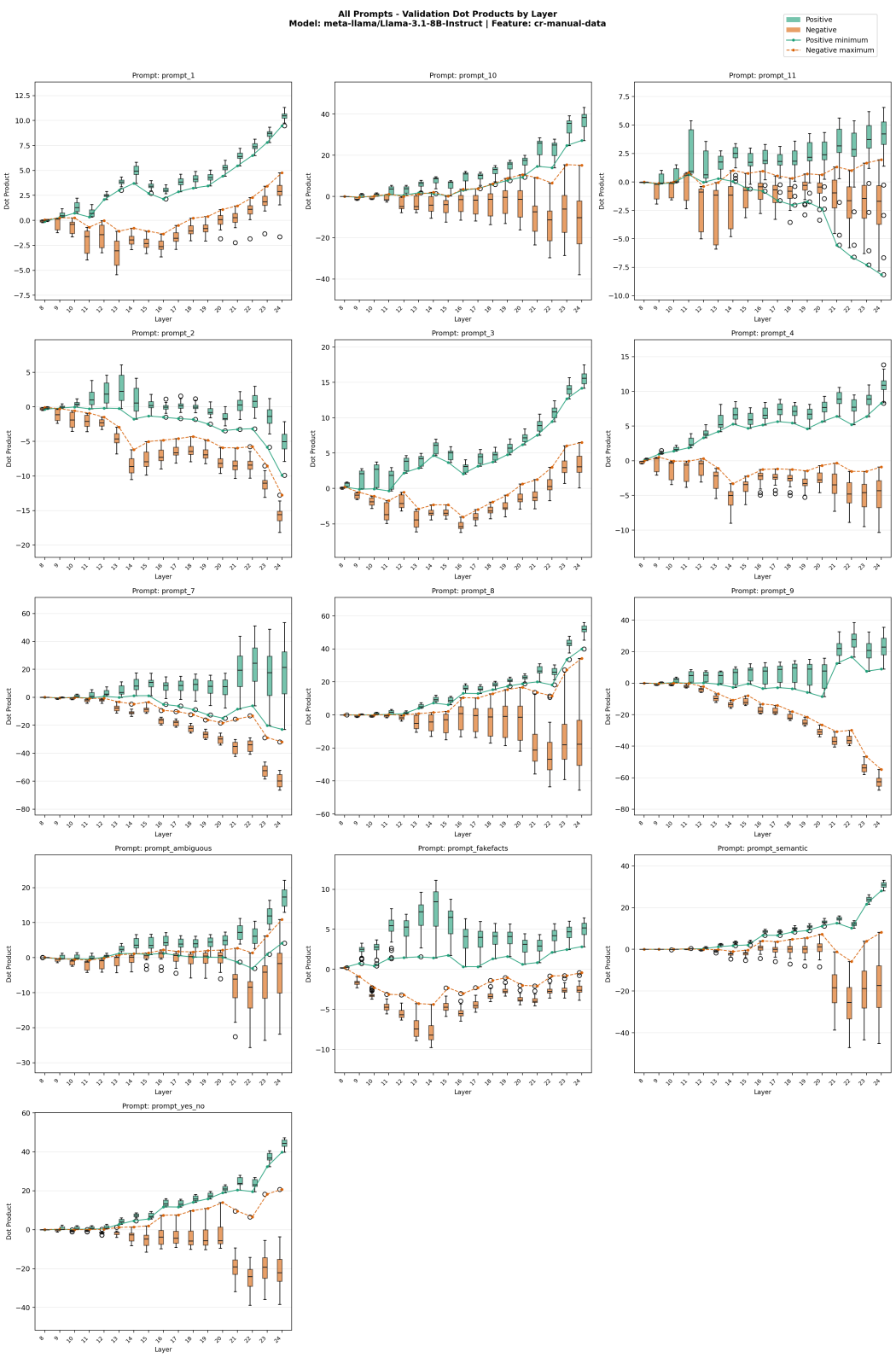


Figure 4: Layer-wise projection of held-out context-reliant (green) and parametric (orange) examples onto prompt-specific steering vectors for Llama-3.1-8B-Instruct, one panel per template. Boxes show the validation distributions; the green and orange curves show the minimum context-reliant and maximum parametric projection, respectively. Templates with an easily reached context answer (`prompt_1`, `prompt_3`, `prompt_4`, `prompt_9`, `prompt_yes_no`) separate cleanly from around layer 14. Templates that embed the parametric answer as a decoy already present in the passage (`prompt_10`, `prompt_11`) remain overlapped until layer 20–21. `prompt_ambiguous` separates similarly late despite having no such decoy, so the delay there has a different cause.

Figure 5 shows the same projection for a single steering vector fit on the pooled 13-template dataset, rather than one vector per template. It illustrates why later layers are not selected despite showing a larger raw gap between classes: the spread of both distributions, and in particular the worst-case outliers, grows substantially from layer 19 onward.

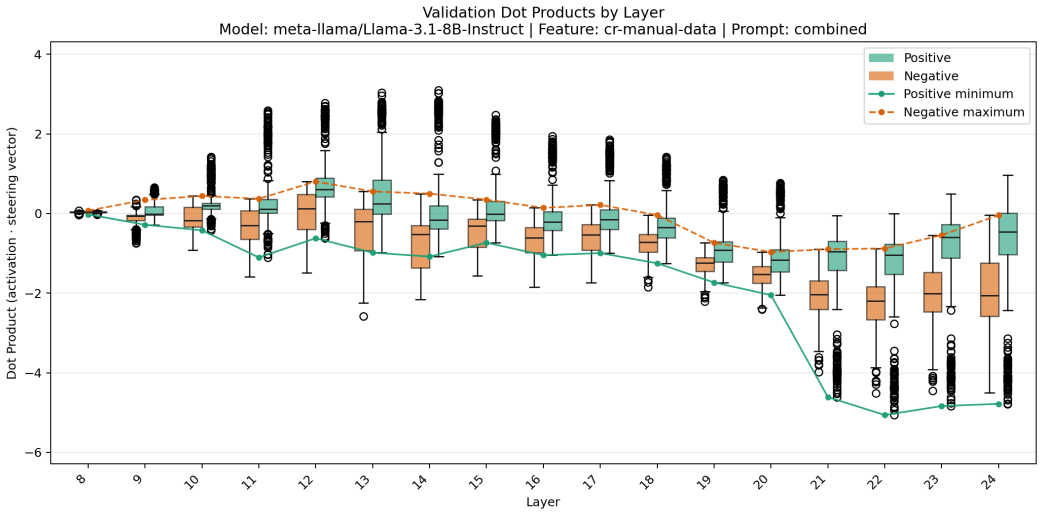


Figure 5: Layer-wise projection of held-out context-reliant (green) and parametric (orange) examples onto a single steering vector fit on the pooled 13-template dataset for Llama-3.1-8B-Instruct. The solid and dashed lines track the positive minimum and negative maximum per layer. Both distributions overlap heavily through layer 18; from layer 19 onward the positive minimum drops sharply (to about -5 by layer 24), showing that the larger apparent gap at later layers comes with substantially more extreme outliers, not a cleaner separation.