Mixture-of-Steering Vectors (MoSV): Sparse Gating for Compositional Hallucination Mitigation

Daniel Lee [*1] **Feolu Kolawole** [*1] **Vedant Srinivas** [*1]

Abstract

Large language models remain prone to hallucination despite advances in scale and training, yet existing inference-time steering methods apply a single global correction vector to every input, treating all hallucinations as one monolithic failure mode. We propose **Mixture-of-Steering Vectors (MoSV)**, which discovers multiple hallucination subspaces from contrastive activation data via unsupervised clustering, then trains a lightweight sparse router to select and compose the appropriate vector(s) per prompt at inference time. Evaluated on DefAn (Rahman et al., 2024), a factual QA benchmark spanning eight structured knowledge domains (n=10,615), MoSV-K8 improves exact-match accuracy from 19.7% (Vanilla) to 22.1% (+2.4pp, p=9.1×10^{-6}), while single-vector CAA yields only a negligible, non-significant gain (+0.3pp). A random-routing ablation, which selects vectors without the learned router, *degrades* accuracy to 17.4% ($-$2.3pp), confirming that per-prompt routing is the operative mechanism. Analysis reveals that K-means clusters of contrastive diff vectors recover ground-truth domain boundaries without any label supervision, providing an unsupervised account of why compositional steering is effective.

1. Introduction

Large language models have achieved strong performance across a wide range of tasks, yet hallucination remains a persistent failure, appearing despite increases in training data or model size. Hallucinations appear in many different forms, including fabricated evidence, numerical errors, and misconceptions, each of which can undermine reliability in real-world safety-sensitive applications. Prior surveys identify hallucination as one of the primary reliability challenges for deploying LLMs in real-world applications, emphasizing that models frequently produce incorrect answers even when relevant knowledge appears to be encoded within their internal representations (Huang et al., 2025; Ji et al., 2023; Anh-Hoang et al., 2025). Because these errors have different sources, methods that help in one case do not always generalize consistently across tasks or domains.

Activation steering offers an alternative strategy by intervening directly in a model's internal activations during inference. Prior work demonstrates that adding a learned vector to the residual stream can influence properties such as toxicity, sentiment, or refusal behavior while preserving performance on unrelated tasks (Turner et al., 2024). Contrastive Activation Addition (CAA) extends this idea by deriving steering vectors from contrastive datasets (Rimsky et al., 2024). However, existing steering methods typically apply a single global vector across all prompts, which may be overly restrictive for hallucination mitigation where distinct failure modes call for distinct corrections.

In this work, we introduce **Mixture-of-Steering Vectors (MoSV)**, a framework that extends CAA by replacing a single global steering direction with a small set of directions selected per prompt. MoSV uses a linear probe to identify the transformer layer where contrastive diff vectors have the most geometric structure, then clusters those differences at that layer to let the model's own geometry determine the groupings. At inference time, a sparse router conditioned on each prompt's hidden representation selects and combines up to two vectors before injecting the result into the residual stream. We evaluate on DefAn (Rahman et al., 2024), a factual QA benchmark covering eight structured knowledge domains, and show that the model's activation geometry naturally separates hallucination types along domain boundaries.

2. Related Work

Existing factual QA benchmarks each have limitations for studying hallucination: TruthfulQA (Lin et al., 2022) re-

[*]Equal contribution [1]Department of Computer Science, Stanford University, Stanford, CA, USA. Correspondence to: Daniel Lee <leedan@stanford.edu>, Feolu Kolawole <flukol@stanford.edu>, Vedant Srinivas <vedants8@stanford.edu>.

Proceedings of the 43rd International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

quires an LLM judge, MMLU (Hendrycks et al., 2021) is multiple-choice only, and SQuAD 2.0 (Rajpurkar et al., 2018) is limited to reading comprehension. We use DefAn (Rahman et al., 2024), which provides short unambiguous factual answers across eight structured domains, enabling exact-match evaluation without any external judge.

Inference-time hallucination mitigation approaches include RAG (Lewis et al., 2021), which requires external retrieval infrastructure, and DoLa (Chuang et al., 2024), which contrasts transformer layers but does not model distinct internal failure modes. Activation steering (Turner et al., 2024) and CAA (Rimsky et al., 2024) intervene directly in the residual stream but apply a single global vector to every prompt. MoSV extends CAA by learning multiple steering directions and routing among them per prompt, drawing on sparse mixture-of-experts gating (Shazeer et al., 2017).

3. Approach

MoSV extends CAA from a single fixed correction to a set of steering vectors selected dynamically per prompt. The pipeline has four stages: building a contrastive training set, extracting activations, clustering those activations into a basis of steering vectors, and training a lightweight router to compose a prompt-specific correction at inference time.

3.1. Dataset and Contrastive Pair Construction

We use DefAn (Rahman et al., 2024), a factual QA benchmark covering eight domains: FIFA World Cup results, US census data, Nobel prizes, Academy Awards, UN founding dates, QS university rankings, academic conference metadata, and arithmetic. Answers are short factual strings (e.g., "France", "1945"), enabling cheap exact-match evaluation without an LLM judge.

We split each domain 85/15 into train and eval *before* any model inference, ensuring zero contamination of the held-out set. LLaMA-3.1-8B-Instruct (Grattafiori et al., 2024) is then run greedily on the train portion. A question is kept if and only if the model's response does not contain the ground-truth answer under case-insensitive normalization. These failures become contrastive training triplets $(q,\, a^+,\, a^-)$, where a^+ is the ground-truth answer and a^- is the model's hallucinated response.

3.2. Contrastive Activation Extraction

For each training triplet, we run two forward passes through LLaMA-3.1-8B-Instruct with hooks attached to eleven candidate layers $\mathcal{L} = \{8, 10, 12, 13, 14, 15, 16, 17, 18, 20, 22\}$, capturing the residual-stream hidden state at the *final token position* of each completion. The contrastive direction for pair i at layer ℓ is:

$$\Delta_i^{(\ell)} = \mathbf{a}_i^{+(\ell)} - \mathbf{a}_i^{-(\ell)} \tag{1}$$

A third, prompt-only forward pass simultaneously collects representations $\mathbf{h}_i^{(\ell)}$ that will serve as router inputs during training. All passes are performed in a single sweep over the training pairs.

3.3. Layer Selection

Rather than fixing the intervention layer by hand, we select it from the data. We fit a logistic regression probe on the diff vectors $\{\Delta_i^{(\ell)}\}$ at each layer in $\mathcal{L}$ and evaluate via five-fold cross-validation. Because the diff vectors carry no inherent binary labels, we assign arbitrary first-half/second-half labels; the resulting accuracy measures how much *geometric structure* the diff vectors possess at each layer. A layer with high accuracy has high-variance, well-separated representations, making it the most information-rich location to inject a corrective signal. We designate ℓ^* as the layer with the highest such score.

3.4. Discovering Multiple Steering Vectors

The central novelty of MoSV is that we do not collapse the N contrastive directions into a single mean vector. Instead, we run K-means on $\{\Delta_i^{(\ell^*)}\}$ (after standard scaling and PCA reduction to 50 components) to obtain K cluster centroids. Each centroid is inverse-transformed back to the original hidden-state space, yielding a bank of steering vectors $\{v_1, \ldots, v_K\}$. The clustering is entirely unsupervised: we never specify what types of hallucinations each cluster should represent, allowing the model's own activation geometry to reveal natural groupings. We evaluate $K \in \{2, 4, 6, 8, 10, 15, 20, 35, 50\}$ and report results for each value; silhouette scores are computed post-hoc as an interpretability metric rather than as a selection criterion.

3.5. Sparse Prompt-Conditioned Router

At inference time we need to select among the K vectors per prompt. Drawing on the sparse-gating formulation of mixture-of-experts models (Shazeer et al., 2017), we train a three-layer MLP with a residual skip connection to map each prompt's hidden representation $\mathbf{h}^{(\ell^*)}$ to a score over the K vectors. To keep the intervention focused, we apply top-κ sparsification ($\kappa = 2$): only the two highest-scored vectors receive nonzero weight, and those weights are renormalized with softmax over the selected pair.

A key failure mode of sparse routing is load collapse, where the router learns to ignore all but one direction. We address

this with a load-balancing regularizer:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} \; + \; \lambda_{\text{bal}} \cdot \text{Var}\left(\frac{1}{B} \sum_{i=1}^{B} \text{softmax}(\mathbf{z}_i) \right) \quad (2)$$

where $\mathbf{z}_i \in \mathbb{R}^K$ are the router logits for sample i and B is the batch size. The variance term penalizes uneven utilization across the K directions within each batch. We set $\lambda_{\text{bal}} = 0.01$ and train with Adam and cosine annealing for 100 epochs.

3.6. Inference-Time Steering

At generation time, the router computes a sparse weighted combination of the K steering vectors from the prompt's hidden state. This composite vector $v_{\text{comp}} = \sum_k w_k v_k$ is fixed for the prompt and injected into the residual stream at every decoding step via a forward hook:

$$\mathbf{h}' = \mathbf{h} + \alpha \cdot v_{\text{comp}} \quad (3)$$

The scalar α controls intervention strength. Computing the composite vector once from the prompt avoids any per-step router overhead.

3.7. Baselines

We compare MoSV against two reference points. **Vanilla** is the unmodified LLaMA-3.1-8B-Instruct model with no intervention. **Single-Vec CAA** computes the global mean of all training diff vectors and applies it uniformly to every prompt (Rimsky et al., 2024), serving as the direct predecessor our method extends. All systems share identical generation parameters and are evaluated with the same exact-match scoring: a response is counted correct if the normalized ground-truth string appears as a substring of the normalized model output.

4. Experiments

4.1. Data

We evaluate on the held-out 15% per-domain split of DefAn (Rahman et al., 2024) described in Section 3.1, comprising 10,615 items across all eight knowledge domains. The split is reserved before any model inference, guaranteeing zero contamination. Each item has a short, verifiable ground-truth answer (typically a name, number, or year), enabling exact-match scoring with no external judge.

4.2. Evaluation Method

We use exact-match accuracy: a response is scored as correct if the normalized ground-truth answer appears as a substring of the normalized model output, where normalization lowercases the string and collapses punctuation and whitespace. This scoring requires no external judge and is fully reproducible.

Table 1. DefAn exact-match accuracy (n=10,615, α=0.5). Δ vs. Vanilla. BH-corrected proportion z-tests (m=11); *** $q<0.05$, ns = not significant. †Significantly *worse* than Vanilla ($p=6.9\times10^{-6}$); routing ablation.

System	Acc.	Δ	p	BH
Vanilla	19.7%	—	—	—
Single-Vec CAA (Rimsky et al., 2024)	20.0%	+0.3pp	0.292	ns
Random-K8 (±0.08pp)	17.4%	−2.3pp	6.9×10^{-6}	†
MoSV-K2	20.8%	+1.1pp	0.028	***
MoSV-K4	21.9%	+2.2pp	5.0×10^{-5}	***
MoSV-K6	21.8%	+2.1pp	8.1×10^{-5}	***
MoSV-K8	**22.1%**	**+2.4pp**	$\mathbf{9.1\times10^{-6}}$	***
MoSV-K10	**22.1%**	**+2.4pp**	$\mathbf{1.1\times10^{-5}}$	***
MoSV-K15	21.8%	+2.1pp	8.1×10^{-5}	***
MoSV-K20	21.5%	+1.8pp	6.7×10^{-4}	***
MoSV-K35	21.4%	+1.7pp	1.0×10^{-3}	***
MoSV-K50	21.6%	+1.9pp	3.1×10^{-4}	***

4.3. Experimental Details

All experiments use LLaMA-3.1-8B-Instruct loaded with 8-bit quantization on a single NVIDIA L40S GPU (46 GB). Contrastive activations are extracted at layers $\mathcal{L} = \{8, 10, 12, 13, 14, 15, 16, 17, 18, 20, 22\}$. PCA is applied to 50 components before K-means clustering ($n_\text{init} = 20$, $\text{random_state} = 42$). We evaluate $K \in \{2, 4, 6, 8, 10, 15, 20, 35, 50\}$ and report results for each value. The router is trained for 100 epochs with Adam (lr $= 10^{-3}$, weight decay $= 10^{-4}$) and cosine annealing, with input dropout 0.3 and load-balance coefficient $\lambda_{\text{bal}} = 0.01$. Steering strength is fixed at $\alpha = 0.5$ for all reported results. Evaluation uses greedy decoding with a maximum of 80 new tokens.

4.4. Results

All MoSV variants with $K \geq 2$ outperform Vanilla and survive Benjamini–Hochberg (BH) correction. Single-Vec CAA provides a negligible, non-significant gain (+0.3pp, $p = 0.29$), demonstrating that the benefit of MoSV comes specifically from the mixture rather than from steering in general. Performance peaks at $K = 8$–10 (+2.4pp) and degrades only modestly at $K = 50$ (+1.9pp), indicating robustness to over-specification of K.

5. Analysis

5.1. Cluster Interpretability

A central question is whether the unsupervised clusters discovered by MoSV correspond to interpretable structure. Figure 1 shows t-SNE projections of a stratified sample of the training diff vectors colored by (a) ground-truth domain and

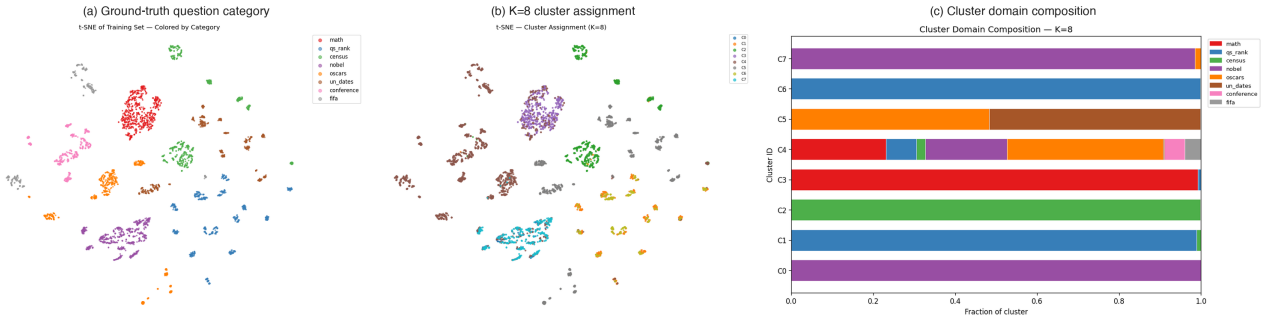

Figure 1. (a) t-SNE of training diff vectors colored by ground-truth domain. (b) Same projection colored by K=8 cluster assignment. (c) Per-cluster domain composition. Clusters (b) closely recover the domain partition (a) without any label supervision.

(b) K=8 cluster assignment, alongside (c) the per-cluster domain composition bar chart. Panels (a) and (b) are strikingly similar: without any domain labels, K-means recovers a partition that closely mirrors the ground-truth domain boundaries. At $K = 8$, six of eight clusters are dominated by a single domain at >98% purity. The two mixed clusters both involve domains with similar surface-form answers (Oscars and UN Dates), where the contrastive directions are geometrically entangled.

This interpretability suggests a practical extension: because each steering vector can be attributed to a specific domain, individual vectors can be selectively disabled or reweighted without retraining the underlying model.

5.2. Effect of K and Relationship to Domain Count

Accuracy rises from $K = 2$ (20.8%) to $K = 4$ (21.9%, +1.1pp) then plateaus, with a total range of only 0.7pp across $K \in \{4, \ldots, 50\}$. The peak at $K = 8$–10 aligns with the eight knowledge domains in DefAn, consistent with the hypothesis that each semantic domain requires approximately one dedicated steering direction. Accuracy differences between $K = 4$ and any larger K are not individually significant, so we do not claim a sharp optimum; rather, the method is robust to over-specification once K exceeds the number of semantic categories in the benchmark.

Silhouette score decreases monotonically with K (from 0.170 at $K = 4$ to 0.063 at $K = 50$), yet accuracy remains flat. The Pearson correlation between silhouette and accuracy across all K is $r = 0.10$ ($p = 0.79$), indicating that clustering geometry and downstream steering effectiveness are largely decoupled. The router compensates for geometric overlap at higher K by learning finer-grained routing assignments.

5.3. Router Load Balance and Routing Ablation

Router validation accuracy decreases predictably with K (K2: 97.2%, K8: 87.1%, K20: 74.5%), consistent with a harder classification problem as clusters become more geometrically similar at higher K. The load-balance regularizer prevents collapse: at $K = 8$, all eight clusters receive non-trivial assignment mass, and no single cluster captures more than 25% of routing decisions on the held-out eval set.

Random routing ablation. To isolate the contribution of the learned router, we evaluate a Random-K8 baseline that selects 2 of the 8 steering vectors uniformly at random at each inference step (mean over 3 seeds). Random routing achieves only 17.4% accuracy, a 2.3pp *decrease* relative to Vanilla and 3.6pp below MoSV-K8 (Table 1). The variance across seeds is negligible (±0.08pp), confirming this is a stable result. This finding has two implications. First, vector injection without routing is actively harmful: presenting the model with the wrong correction direction degrades factual accuracy. Second, the router's per-prompt selectivity is the operative mechanism behind MoSV's gains, not the mere act of adding vectors to the residual stream. The 3.6pp gap between random and learned routing empirically validates the routing component as a necessary contribution.

6. Conclusion

We proposed MoSV, an inference-time steering framework that replaces a single global correction vector with a bank of K specialized vectors discovered via unsupervised clustering of contrastive activation differences. A sparse MLP router selects which vectors to apply per prompt, conditioned on the prompt's hidden representation. Evaluated on DefAn ($n = 10{,}615$) across eight factual domains, MoSV-K8 achieves 22.1% exact-match accuracy versus 19.7% for Vanilla (+2.4pp, $p = 9.1 \times 10^{-6}$) and 20.0% for Single-Vec CAA (+0.3pp, $p = 0.29$). The central empirical finding is that a single steering direction provides no meaningful

benefit, while a learned mixture does. A random-routing ablation further confirms that the learned router — not vector injection in general — is the operative mechanism.

Limitations and future work include: extending evaluation to open-domain benchmarks where domain boundaries are less crisp, investigating whether the domain-cluster alignment generalizes to other model families and factual datasets, and exploring routing architectures with finer-grained per-domain control.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning, specifically the reliability of large language models through improved hallucination mitigation. The primary societal benefit is more factually accurate language model outputs, which reduces the risk of misinformation in downstream applications. We do not foresee negative societal consequences specific to this work beyond those generally associated with advances in LLM capability.

References

Anh-Hoang, D., Tran, V., and Nguyen, L.-M. Survey and analysis of hallucinations in large language models: attribution to prompting strategies or model behavior. *Frontiers in Artificial Intelligence*, Volume 8 - 2025, 2025. ISSN 2624-8212. doi: 10.3389/frai.2025.1622292. URL https://www.frontiersin.org/journals/artificial-intelligence/articles/10.3389/frai.2025.1622292.

Chuang, Y.-S., Xie, Y., Luo, H., Kim, Y., Glass, J., and He, P. Dola: Decoding by contrasting layers improves factuality in large language models, 2024. URL https://arxiv.org/abs/2309.03883.

Grattafiori, A. et al. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding, 2021. URL https://arxiv.org/abs/2009.03300.

Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., and Liu, T. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55, January 2025. ISSN 1558-2868. doi: 10.1145/3703155. URL http://dx.doi.org/10.1145/3703155.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, March 2023. ISSN 1557-7341. doi: 10.1145/3571730. URL http://dx.doi.org/10.1145/3571730.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., tau Yih, W., Rocktäschel, T., Riedel, S., and Kiela, D. Retrieval-augmented generation for knowledge-intensive nlp tasks, 2021. URL https://arxiv.org/abs/2005.11401.

Lin, S., Hilton, J., and Evans, O. Truthfulqa: Measuring how models mimic human falsehoods, 2022. URL https://arxiv.org/abs/2109.07958.

Rahman, A. B. M. A., Anwar, S., Usman, M., and Mian, A. Defan: Definitive answer dataset for llms hallucination evaluation, 2024. URL https://arxiv.org/abs/2406.09155.

Rajpurkar, P., Jia, R., and Liang, P. Know what you don't know: Unanswerable questions for SQuAD. In *Association for Computational Linguistics (ACL)*, 2018.

Rimsky, N., Gabrieli, N., Schulz, J., Tong, M., Hubinger, E., and Turner, A. M. Steering llama 2 via contrastive activation addition, 2024. URL https://arxiv.org/abs/2312.06681.

Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., and Dean, J. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations (ICLR)*, 2017. URL https://arxiv.org/abs/1701.06538.

Turner, A. M., Thiergart, L., Leech, G., Udell, D., Vazquez, J. J., Mini, U., and MacDiarmid, M. Steering language models with activation engineering, 2024. URL https://arxiv.org/abs/2308.10248.