Do Function Vectors Factor Task and Distribution?

Essam Sleiman[1] **Jacob Andreas**[2]

[1]Harvard University [2]Massachusetts Institute of Technology

Abstract

Function vectors provide compact activation-space representations of in-context learning behavior, but it remains unclear whether they encode abstract task structure independently of the demonstrations used to construct them. Since in-context demonstrations specify not only an input-output mapping but also a label space, format, and input distribution, function vectors may inherit distribution-specific information. We test this possibility using controlled categorical tasks that share a similar category-selection structure but differ in their word distributions. We construct task-conditioned function vectors and distribution-only vectors, then evaluate whether simple vector arithmetic can adapt a function vector from one distribution to another. We find that function-vector behavior is highly sensitive to intervention scale. We also find that adding a target-distribution vector can partially recover an out-of-distribution transfer gap, but subtracting the source-distribution vector does not restore in-distribution performance. These results suggest that function vectors are distribution-sensitive representations of in-context behavior rather than cleanly factorized task vectors.

1. Introduction

In-context learning allows language models to perform new tasks from demonstrations without parameter updates. However, demonstrations do not specify only the task mapping. Prior work shows that in-context performance also depends on the label space, input distribution, and prompt format (Min et al., 2022). This makes it difficult to determine whether a model is learning an abstract task, recognizing a familiar distribution, or relying on a mixture of both.

Function vectors make this question testable inside the model. Recent work shows that task behavior induced by in-context demonstrations can be extracted from model activations into a compact vector and added back during inference to induce the corresponding behavior without demonstrations (Todd et al., 2024). But because function vectors are extracted from demonstration-conditioned activations, they may encode both task-relevant and distribution-specific information.

We ask whether function vectors factor into separable task and distribution components. We study controlled categorical tasks such as selecting a color from a list of animals or selecting a verb from a list of adjectives. These tasks share a similar category-selection structure but differ in their lexical distributions. We construct task-conditioned function vectors, construct distribution-only vectors by removing task structure while preserving distributional tokens, and test whether arithmetic over these vectors improves out-of-distribution transfer.

Our results support a cautious conclusion. Function vectors are clearly distribution-sensitive: a vector constructed on one distribution transfers substantially worse to another. Adding a target-distribution vector partially recovers this gap, but subtracting the source-distribution vector is inconsistent and does not restore in-distribution performance. We also find that intervention scale strongly changes behavior. Together, these results suggest that function vectors are useful activation-space interventions, but not cleanly factorized task representations.

This paper makes three contributions. First, we formulate a controlled test of whether function vectors factor task and distribution information. Second, we construct distribution-only vectors and use them to probe whether lexical-distribution information can be added to or removed from function vectors. Third, we show that both arithmetic composition and intervention scale matter: target-distribution addition can partially recover OOD performance, but source-distribution subtraction and naive averaging do not reliably isolate a distribution-invariant task vector.

2. Related Work

Prior work on in-context learning shows that demonstrations provide several signals beyond input-output mappings. Min et al. (2022) show that label space, input distribution, and prompt format are central to in-context learning performance. Pan et al. (2023) distinguish task recognition from task learning, and Wei et al. (2023) study how models trade off semantic priors against in-context mappings. This motivates our question: if function vectors are extracted from demonstration-conditioned activations, do they encode a clean task representation or a mixture of task and distribution information?

Function vectors provide a compact representation of in-context task behavior. Todd et al. (2024) identify attention heads with strong causal effects on in-context predictions, average their task-conditioned activations, and show that adding the resulting vector can induce task behavior at inference time. They also show that function vectors support some semantic composition, while leaving open what information besides output space is encoded. Related work on in-context vectors similarly constructs latent vectors from demonstrations and studies magnitude control and vector arithmetic (Liu et al., 2024).

Our work also relates to activation-space steering methods such as Representation Engineering, Activation Addition, and Contrastive Activation Addition (Zou et al., 2023; Turner et al., 2023; Rimsky et al., 2024). These methods show that activation-space directions can steer model behavior at inference time. In contrast, we focus specifically on function vectors extracted from in-context demonstrations and ask whether their induced behavior transfers across lexical distributions through simple arithmetic.

The closest prior result for our purposes is the observation that function vectors can sometimes be composed through vector arithmetic. We study a different aspect of compositionality. Rather than composing two task behaviors, we ask whether a task-like behavior can be separated from the distribution that elicited it. This distinction matters because in-context demonstrations are not distribution-neutral: they specify the words, categories, formatting, and output space from which a function vector is extracted.

3. Method

We use controlled categorical tasks from the function-vector setup, including color_v_animal_3 and verb_v_adjective_3. In color_v_animal_3, the model selects a color from a list containing animals; in verb_v_adjective_3, it selects a verb from a list containing adjectives. The tasks share a similar category-selection structure but differ in semantic categories and lexical distributions.

Table 1. Example inputs and outputs for the two main task distributions.

Dataset	Example input	Output
Color/Animal	rhinoceros, woodpecker, black.	black
Verb/Adjective	fearless, climb, light.	climb

For each dataset D, we construct function vectors from 100 randomly sampled 10-shot prompts from the training split. Following prior work, we average the activations of selected attention heads at the final token position:

$$\bar{a}^{D}_{\ell,j} = \frac{1}{|P_D|} \sum_{p \in P_D} a_{\ell,j}(p), \tag{1}$$

$$\mathrm{FV}_D = \sum_{(\ell,j)\in A} \bar{a}^{D}_{\ell,j}. \tag{2}$$

At inference time, we add the vector to the model activation:

$$h' = h + \alpha v, \tag{3}$$

where α is the intervention scale.

To test for distributional information, we also construct a distribution-only vector DV_D for each dataset. These vectors are built using the same activation-averaging procedure, but on prompts that preserve the lexical items from the dataset while removing the input-output task structure. For example, the distribution-only prompts retain words from the color/animal or verb/adjective distributions but remove the demonstration format that maps an input list to the target output. We treat DV_D as an operational proxy for distributional information, not as a guaranteed ground-truth distribution component.

This distinction is important. Our goal is not to assume that DV_D perfectly isolates a distribution direction; instead, we use it to test whether a simple linear factorization is plausible. If a function vector were approximately decomposable as task plus distribution, then replacing the source distribution proxy with the target distribution proxy should improve transfer. If the replacement fails, this suggests that task and distribution information are either not linearly separated by this construction or are entangled in the model representation itself.

For a source distribution S and target distribution T, we compare four interventions:

$$\mathrm{FV}_T \quad \text{in-distribution reference,} \tag{4}$$

$$\mathrm{FV}_S \quad \text{OOD source vector,} \tag{5}$$

$$\mathrm{FV}_S + \mathrm{DV}_T \quad \text{target-distribution addition,} \tag{6}$$

$$\mathrm{FV}_S - \mathrm{DV}_S + \mathrm{DV}_T \quad \text{distribution replacement.} \tag{7}$$

If function vectors decomposed cleanly into task and distribution components, replacing the source distribution vector with the target distribution vector should improve transfer. The first two interventions establish the in-distribution reference and the OOD source-vector baseline. The third tests whether target-distribution information is useful when simply added. The fourth tests the stronger replacement hypothesis.

We evaluate each intervention on zero-shot target queries using average correct-token probability and top-k accuracy for $k \in \{1, 5, 10\}$. All main experiments use GPT-2 XL (Radford et al., 2019). Following the original function-vector setup, we apply the intervention at layer 22 and use the selected causal heads to construct both function vectors and distribution vectors.

4. Results

4.1. Function-Vector Behavior Is Scale-Sensitive

Intervention scale substantially changes function-vector behavior. On color_v_animal_3, the in-distribution function vector continues improving as scale increases, reaching top-1 accuracy of 0.79 around $\alpha = 6.5$, compared with 0.21 at $\alpha = 1.0$ and 0.10 with no intervention. In contrast, verb_v_adjective_3 peaks much earlier, reaching approximately 0.50 top-1 accuracy around $\alpha = 1.5$–2.5 before declining at larger scales.

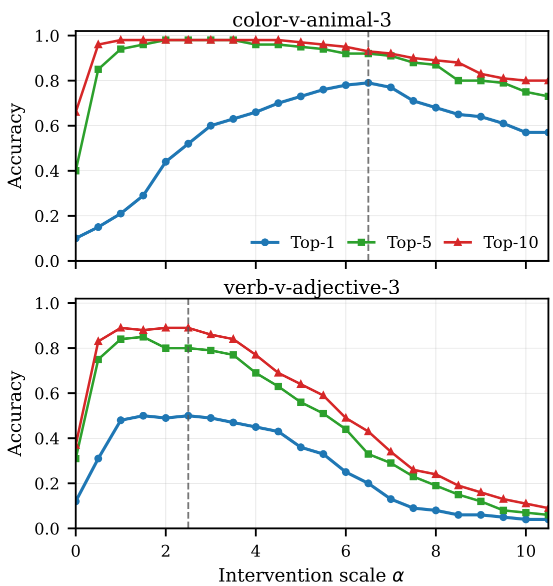

Figure 1. Scaling sensitivity of function vectors. Top-1, top-5, and top-10 accuracy vary substantially with intervention scale, and the best scale differs across datasets.

This shows that raw function-vector magnitude is not directly comparable across datasets. Scale is therefore part of the intervention, not a neutral implementation detail. We also did not observe a simple relationship between optimal scale and vector norm, suggesting that scale calibration may require validation or a more model-aware normalization procedure. Full scaling sweeps are included in Appendix A.

4.2. Distribution Arithmetic Partially Recovers OOD Transfer

We next test whether distribution-vector arithmetic improves transfer from verb_v_adjective_3 to color_v_animal_3. With scale $\alpha = 2.5$, the in-distribution reference $\mathrm{FV}_{\mathrm{c-a}}$ reaches 0.52 top-1 accuracy on color_v_animal_3, while the OOD source vector $\mathrm{FV}_{\mathrm{v-a}}$ reaches only 0.27. This establishes a 0.25-point transfer gap.

Table 2. Main arithmetic results on color/animal, scaled by 2.5.

Intervention	Avg.	T1	T5	T10	Gap
$\mathrm{FV}_{\mathrm{c-a}}$ ID reference	.30	.52	.98	.98	100%
$\mathrm{FV}_{\mathrm{v-a}}$ OOD source	.1632	.27	.83	.90	0%
$\mathrm{FV}_{\mathrm{v-a}} + \mathrm{DV}_{\mathrm{c-a}}$	.20	.39	.85	.95	48%
$\mathrm{FV}_{\mathrm{v-a}} - \mathrm{DV}_{\mathrm{v-a}} + \mathrm{DV}_{\mathrm{c-a}}$	.17	.29	.86	.92	8%

Adding the target distribution vector partially recovers the transfer gap: $\mathrm{FV}_{\mathrm{v-a}} + \mathrm{DV}_{\mathrm{c-a}}$ improves top-1 accuracy from 0.27 to 0.39, recovering 48% of the gap to the in-distribution reference. This suggests that distributional information can influence the behavior induced by a function vector.

However, the stronger replacement intervention does not support clean factorization. If $\mathrm{FV}_{\mathrm{v-a}}$ were approximately a sum of an independent task component and a source-distribution component, then $\mathrm{FV}_{\mathrm{v-a}} - \mathrm{DV}_{\mathrm{v-a}} + \mathrm{DV}_{\mathrm{c-a}}$ should outperform simple target-distribution addition. Instead, it reaches only 0.29 top-1 accuracy. This suggests that subtracting the source distribution vector removes or disrupts useful information, and that task and distribution information are entangled rather than cleanly linearly separable.

These results also clarify what the positive result does and does not show. The target-distribution vector appears to provide useful information about the target lexical space, but it does not turn the source function vector into the target function vector. This is why we interpret the improvement as evidence of distribution sensitivity rather than as evidence that we have recovered a clean task vector.

4.3. Additional Variants Reinforce the Same Pattern

Additional unscaled, normalization, and averaging experiments show similarly mixed behavior. Without scaling, target-distribution arithmetic is much less informative: on color_v_animal_3, the OOD source vector reaches 0.21 top-1 accuracy, while distribution replacement reaches 0.19. This weakens the idea that subtraction and addition alone produce a stable transfer operation.

We also test a normalization variant in which the function vector and activation are normalized before addition. This does not remove the qualitative instability of the arithmetic interventions. On color_v_animal_3, target-DV addition improves the OOD source vector from 0.24 to 0.37 top-1 accuracy, but replacement reaches only 0.27. On verb_v_adjective_3, adding the target distribution vector to the Color/Animal source vector hurts performance, dropping top-1 accuracy from 0.20 to 0.09. Simple norm correction is therefore not enough to make function-vector arithmetic robust.

Finally, we test whether averaging function vectors from multiple distributions produces a more distribution-invariant task vector. It does not reliably help: averaged vectors generally perform similarly to or worse than individual source vectors, except when the average includes an in-distribution component. This is consistent with the main result: simple vector operations can sometimes improve performance, but they do not reliably isolate a clean task representation. Full results are included in Appendices B–E.

Because top-5 and top-10 accuracy are often near ceiling, we focus on top-1 accuracy and average correct-token probability. Top-k metrics remain useful diagnostics because they show when interventions place the correct token in the high-probability candidate set without making it the top prediction. The factorization question is most clearly reflected in whether the intervention makes the correct token the model's preferred output.

5. Discussion

The scale-sensitivity result suggests that a function vector should be understood as both a direction and an intervention magnitude. Two vectors that appear comparable as task representations can have very different effects when added to the residual stream at the same scale. In our experiments, color_v_animal_3 continues improving at scales where verb_v_adjective_3 has already begun to degrade. This means that raw vector magnitude is not naturally calibrated across tasks, and that evaluating a function vector at a single arbitrary scale can obscure its actual behavior.

The arithmetic results also suggest that distributional information is not merely an additive nuisance component. If the source distribution vector could be subtracted cleanly, then $\mathrm{FV}_S - \mathrm{DV}_S + \mathrm{DV}_T$ should be the most direct adaptation from source to target. Instead, this replacement is weaker than simple target-DV addition. One interpretation is that the distribution-only vector captures information that overlaps with task execution, so subtracting it removes useful structure. Another is that the function vector itself represents the task in a category-specific way: for example, "select the color" may not be represented as the same operation as "select the verb," even if both tasks share an abstract minority-category structure.

These findings point to a broader caution for activation-space arithmetic. A vector can be behaviorally useful without being compositionally clean. Function vectors may successfully steer a model toward an in-context behavior, while still encoding a mixture of task, distribution, output-space, and formatting information. This suggests that future work should evaluate activation-space interventions not only by their in-distribution steering performance, but also by their stability under scale changes and their transfer behavior across related distributions.

6. Conclusion and Limitations

We tested whether function vectors factor into separable task and distribution components. Our results suggest that they do not, at least under simple linear arithmetic. Function vectors are distribution-sensitive, target-distribution addition can partially recover an OOD transfer gap, and source-distribution subtraction does not restore in-distribution performance. Intervention scale also strongly affects behavior, with different datasets requiring different scales for best performance.

These findings suggest that function vectors are useful but entangled representations of in-context behavior. They can steer model behavior, but their effect depends on the construction distribution, the intervention scale, and the semantic categories used in the demonstrations. Future work should develop better methods for calibrating intervention scale, constructing distribution vectors, and testing whether similar patterns hold across larger models, more realistic tasks, different intervention layers, and multi-layer interventions.

This study is intentionally small and controlled. Our main experiments use GPT-2 XL and synthetic categorical datasets; our distribution vectors are proxy constructions rather than ground-truth distribution components; and our scale choices are exploratory rather than learned from a held-out validation procedure. The results should be read as an empirical probe of function-vector factorization, not as a general method for robust OOD adaptation.

Acknowledgments. This work was conducted as part of MIT 6.8610, Quantitative Methods for Natural Language Processing, Fall 2023.

References

Liu, S., Ye, H., Xing, L., and Zou, J. Y. In-context vectors: Making in context learning more effective and controllable through latent space steering. In *International Conference on Machine Learning*, 2024.

Min, S., Lyu, X., Holtzman, A., Artetxe, M., Lewis, M., Hajishirzi, H., and Zettlemoyer, L. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022.

Pan, J., Gao, T., Chen, H., and Chen, D. What in-context learning "learns" in-context: Disentangling task recognition and task learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, 2023.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners, 2019. OpenAI technical report.

Rimsky, N., Gabrieli, N., Schulz, J., Tong, M., Hubinger, E., and Turner, A. M. Steering llama 2 via contrastive activation addition, 2024. arXiv preprint arXiv:2312.06681.

Todd, E., Li, M., Sharma, A. S., Mueller, A., Wallace, B. C., and Bau, D. Function vectors in large language models. In *International Conference on Learning Representations*, 2024.

Turner, A. M., Thiergart, L., Udell, D., Leech, G., Mini, U., and MacDiarmid, M. Activation addition: Steering language models without optimization, 2023. arXiv preprint arXiv:2308.10248.

Wei, J., Wei, J., Tay, Y., Tran, D., Webson, A., Lu, Y., Chen, X., Liu, H., Huang, D., Zhou, D., and Ma, T. Larger language models do in-context learning differently, 2023. arXiv preprint arXiv:2303.03846.

Zou, A., Phan, L., Chen, S., Campbell, J., Guo, P., Ren, R., Pan, A., Yin, X., Mazeika, M., Dombrowski, A.-K., Goel, S., Li, N., Byun, M. J., Wang, Z., Mallen, A., Basart, S., Koyejo, S., Song, D., Fredrikson, M., Kolter, J. Z., and Hendrycks, D. Representation engineering: A top-down approach to ai transparency, 2023. arXiv preprint arXiv:2310.01405.

A. Scaling Sweeps

Table 3. Scaling sweeps for the two main task distributions.

A.1 color_v_animal_3

Scale	Avg.	T1	T5	T10
0.0	0.020	0.10	0.40	0.66
0.5	0.077	0.15	0.85	0.96
1.0	0.130	0.21	0.94	0.98
1.5	0.183	0.29	0.96	0.98
2.0	0.244	0.44	0.98	0.98
2.5	0.304	0.52	0.98	0.98
3.0	0.357	0.60	0.98	0.98
3.5	0.404	0.63	0.98	0.98
4.0	0.443	0.66	0.96	0.98
4.5	0.476	0.70	0.96	0.98
5.0	0.502	0.73	0.95	0.97
5.5	0.524	0.76	0.94	0.96
6.0	0.536	0.78	0.92	0.95
6.5	0.538	0.79	0.92	0.93
7.0	0.530	0.77	0.91	0.92
7.5	0.513	0.71	0.88	0.90
8.0	0.490	0.68	0.87	0.89
8.5	0.464	0.65	0.80	0.88
9.0	0.435	0.64	0.80	0.83
9.5	0.408	0.61	0.79	0.81
10.0	0.384	0.57	0.75	0.80
10.5	0.362	0.57	0.73	0.80

A.2 verb_v_adjective_3

Scale	Avg.	T1	T5	T10
0.0	0.016	0.12	0.31	0.37
0.5	0.068	0.31	0.75	0.83
1.0	0.146	0.48	0.84	0.89
1.5	0.220	0.50	0.85	0.88
2.0	0.274	0.49	0.80	0.89
2.5	0.306	0.50	0.80	0.89
3.0	0.320	0.49	0.79	0.86
3.5	0.317	0.47	0.77	0.84
4.0	0.301	0.45	0.69	0.77
4.5	0.280	0.43	0.63	0.69
5.0	0.246	0.36	0.56	0.64
5.5	0.202	0.33	0.51	0.59
6.0	0.155	0.25	0.44	0.49
6.5	0.112	0.20	0.33	0.43
7.0	0.078	0.13	0.29	0.34
7.5	0.055	0.09	0.23	0.26
8.0	0.041	0.08	0.19	0.24
8.5	0.031	0.06	0.15	0.19
9.0	0.025	0.06	0.12	0.16
9.5	0.021	0.05	0.08	0.13
10.0	0.018	0.04	0.07	0.11
10.5	0.015	0.04	0.06	0.09

B. Unscaled Arithmetic Results

Table 4. Arithmetic results without scaling.

Dataset	Intervention	Avg. prob	Top-1	Top-5	Top-10
Color/Animal	$\mathrm{FV}_{\mathrm{c-a}}$	.16	.25	.95	1.00
Verb/Adjective	$\mathrm{FV}_{\mathrm{v-a}}$	.145	.47	.84	.90
Color/Animal	$\mathrm{FV}_{\mathrm{v-a}}$	.114	.21	.94	.98
Color/Animal	$\mathrm{DV}_{\mathrm{v-a}}$	.14	.21	.96	.98
Color/Animal	$\mathrm{FV}_{\mathrm{v-a}} - \mathrm{DV}_{\mathrm{v-a}} + \mathrm{DV}_{\mathrm{c-a}}$	.12	.19	.94	.98

C. Normalization Variant

We tested a simple normalization procedure in which the function vector and activation were normalized before addition. This did not remove the qualitative instability of the arithmetic interventions.

Table 5. Arithmetic results with a normalization variant.

Dataset	Intervention	Avg. prob	Top-1	Top-5	Top-10
Color/Animal	$\mathrm{FV}_{\mathrm{c-a}}$	0.310	0.530	0.980	0.980
Color/Animal	$\mathrm{FV}_{\mathrm{v-a}}$	0.162	0.240	0.840	0.930
Color/Animal	$\mathrm{FV}_{\mathrm{v-a}} + \mathrm{DV}_{\mathrm{c-a}}$	0.199	0.370	0.930	0.980
Color/Animal	$\mathrm{FV}_{\mathrm{v-a}} - \mathrm{DV}_{\mathrm{v-a}} + \mathrm{DV}_{\mathrm{c-a}}$	0.171	0.270	0.850	0.940
Color/Animal	$\mathrm{FV}_{\mathrm{v-a}} - \mathrm{DV}_{\mathrm{v-a}}$	0.089	0.200	0.780	0.840
Verb/Adjective	$\mathrm{FV}_{\mathrm{v-a}}$	0.292	0.500	0.840	0.890
Verb/Adjective	$\mathrm{FV}_{\mathrm{c-a}}$	0.086	0.200	0.590	0.710
Verb/Adjective	$\mathrm{FV}_{\mathrm{c-a}} + \mathrm{DV}_{\mathrm{v-a}}$	0.040	0.090	0.440	0.530
Verb/Adjective	$\mathrm{FV}_{\mathrm{c-a}} - \mathrm{DV}_{\mathrm{c-a}} + \mathrm{DV}_{\mathrm{v-a}}$	0.066	0.160	0.540	0.640
Verb/Adjective	$\mathrm{FV}_{\mathrm{c-a}} - \mathrm{DV}_{\mathrm{c-a}}$	0.072	0.210	0.580	0.700

D. Averaging Function Vectors

We tested whether averaging function vectors from two distributions improves transfer to a third distribution. Averaging did not reliably produce a distribution-invariant task vector.

Table 6: Averaging function vectors across distributions.

Dataset	Intervention	Avg. prob	Top-1	Top-5	Top-10
Color/Animal	$\mathrm{FV}_{\mathrm{c-a}}$	0.129	0.21	0.95	0.98
Color/Animal	$\mathrm{FV}_{\mathrm{o-c}}$	0.106	0.12	0.92	0.98
Color/Animal	$\mathrm{FV}_{\mathrm{v-a}}$	0.117	0.17	0.93	0.98
Color/Animal	$(\mathrm{FV}_{\mathrm{v-a}} + \mathrm{FV}_{\mathrm{o-c}})/2$	0.112	0.16	0.95	0.98
Color/Animal	$(\mathrm{FV}_{\mathrm{v-a}} + \mathrm{FV}_{\mathrm{c-a}})/2$	0.124	0.19	0.95	0.98
Color/Animal	$(\mathrm{FV}_{\mathrm{o-c}} + \mathrm{FV}_{\mathrm{c-a}})/2$	0.117	0.17	0.95	0.98
Object/Concept	$\mathrm{FV}_{\mathrm{o-c}}$	0.159	0.43	0.78	0.79
Object/Concept	$\mathrm{FV}_{\mathrm{c-a}}$	0.126	0.36	0.76	0.78
Object/Concept	$\mathrm{FV}_{\mathrm{v-a}}$	0.134	0.32	0.75	0.78
Object/Concept	$(\mathrm{FV}_{\mathrm{c-a}} + \mathrm{FV}_{\mathrm{v-a}})/2$	0.132	0.30	0.76	0.78
Object/Concept	$(\mathrm{FV}_{\mathrm{c-a}} + \mathrm{FV}_{\mathrm{o-c}})/2$	0.143	0.40	0.77	0.79
Object/Concept	$(\mathrm{FV}_{\mathrm{v-a}} + \mathrm{FV}_{\mathrm{o-c}})/2$	0.147	0.39	0.78	0.79
Verb/Adjective	$\mathrm{FV}_{\mathrm{v-a}}$	0.148	0.49	0.84	0.89
Verb/Adjective	$\mathrm{FV}_{\mathrm{c-a}}$	0.082	0.30	0.73	0.80
Verb/Adjective	$\mathrm{FV}_{\mathrm{o-c}}$	0.105	0.38	0.76	0.82
Verb/Adjective	$(\mathrm{FV}_{\mathrm{o-c}} + \mathrm{FV}_{\mathrm{c-a}})/2$	0.093	0.31	0.73	0.80
Verb/Adjective	$(\mathrm{FV}_{\mathrm{o-c}} + \mathrm{FV}_{\mathrm{v-a}})/2$	0.124	0.42	0.81	0.85
Verb/Adjective	$(\mathrm{FV}_{\mathrm{v-a}} + \mathrm{FV}_{\mathrm{c-a}})/2$	0.111	0.37	0.79	0.83

E. Additional Arithmetic Results Across Distributions

Table 7: Additional arithmetic results across distributions.

Dataset	Intervention	Avg. prob	Top-1	Top-5	Top-10
Color/Animal	$\mathrm{FV}_{\mathrm{c-a}}$	0.1302	0.21	0.95	0.98
Color/Animal	$\mathrm{FV}_{\mathrm{v-a}}$	0.1169	0.18	0.94	0.98
Color/Animal	$\mathrm{FV}_{\mathrm{v-a}} + \mathrm{DV}_{\mathrm{c-a}}$	0.1417	0.21	0.96	0.98
Color/Animal	$\mathrm{FV}_{\mathrm{v-a}} - \mathrm{DV}_{\mathrm{v-a}} + \mathrm{DV}_{\mathrm{c-a}}$	0.1201	0.18	0.94	0.98
Color/Animal	$\mathrm{FV}_{\mathrm{v-a}} - \mathrm{DV}_{\mathrm{v-a}}$	0.0854	0.18	0.84	0.95
Color/Animal	$\mathrm{FV}_{\mathrm{o-c}}$	0.1067	0.12	0.92	0.98
Color/Animal	$\mathrm{FV}_{\mathrm{o-c}} + \mathrm{DV}_{\mathrm{c-a}}$	0.1276	0.16	0.95	0.98
Color/Animal	$\mathrm{FV}_{\mathrm{o-c}} - \mathrm{DV}_{\mathrm{o-c}} + \mathrm{DV}_{\mathrm{c-a}}$	0.1114	0.11	0.94	0.98
Color/Animal	$\mathrm{FV}_{\mathrm{o-c}} - \mathrm{DV}_{\mathrm{o-c}}$	0.0791	0.15	0.82	0.94
Color/Animal	$\mathrm{FV}_{\mathrm{a-o}}$	0.1071	0.11	0.94	0.97
Color/Animal	$\mathrm{FV}_{\mathrm{a-o}} + \mathrm{DV}_{\mathrm{c-a}}$	0.1251	0.16	0.96	0.98
Color/Animal	$\mathrm{FV}_{\mathrm{a-o}} - \mathrm{DV}_{\mathrm{a-o}} + \mathrm{DV}_{\mathrm{c-a}}$	0.1102	0.11	0.95	0.97
Color/Animal	$\mathrm{FV}_{\mathrm{a-o}} - \mathrm{DV}_{\mathrm{a-o}}$	0.0806	0.13	0.82	0.94
Verb/Adjective	$\mathrm{FV}_{\mathrm{v-a}}$	0.1486	0.50	0.84	0.89
Verb/Adjective	$\mathrm{FV}_{\mathrm{c-a}}$	0.0834	0.30	0.73	0.79
Verb/Adjective	$\mathrm{FV}_{\mathrm{c-a}} + \mathrm{DV}_{\mathrm{v-a}}$	0.0605	0.19	0.61	0.73
Verb/Adjective	$\mathrm{FV}_{\mathrm{c-a}} - \mathrm{DV}_{\mathrm{c-a}} + \mathrm{DV}_{\mathrm{v-a}}$	0.0781	0.27	0.71	0.79
Verb/Adjective	$\mathrm{FV}_{\mathrm{c-a}} - \mathrm{DV}_{\mathrm{c-a}}$	0.0781	0.27	0.73	0.77
Verb/Adjective	$\mathrm{FV}_{\mathrm{a-o}}$	0.0939	0.33	0.75	0.80
Verb/Adjective	$\mathrm{FV}_{\mathrm{a-o}} + \mathrm{DV}_{\mathrm{v-a}}$	0.0697	0.26	0.64	0.76
Verb/Adjective	$\mathrm{FV}_{\mathrm{a-o}} - \mathrm{DV}_{\mathrm{a-o}} + \mathrm{DV}_{\mathrm{v-a}}$	0.0888	0.32	0.73	0.79
Verb/Adjective	$\mathrm{FV}_{\mathrm{a-o}} - \mathrm{DV}_{\mathrm{a-o}}$	0.0853	0.32	0.76	0.79
Verb/Adjective	$\mathrm{FV}_{\mathrm{o-c}}$	0.1048	0.38	0.76	0.82
Verb/Adjective	$\mathrm{FV}_{\mathrm{o-c}} + \mathrm{DV}_{\mathrm{v-a}}$	0.0790	0.28	0.71	0.75
Verb/Adjective	$\mathrm{FV}_{\mathrm{o-c}} - \mathrm{DV}_{\mathrm{o-c}} + \mathrm{DV}_{\mathrm{v-a}}$	0.1036	0.34	0.76	0.82
Verb/Adjective	$\mathrm{FV}_{\mathrm{o-c}} - \mathrm{DV}_{\mathrm{o-c}}$	0.0978	0.33	0.76	0.82
Object/Concept	$\mathrm{FV}_{\mathrm{o-c}}$	0.1593	0.44	0.78	0.79
Object/Concept	$\mathrm{FV}_{\mathrm{c-a}}$	0.1284	0.36	0.76	0.78
Object/Concept	$\mathrm{FV}_{\mathrm{c-a}} + \mathrm{DV}_{\mathrm{o-c}}$	0.1149	0.35	0.76	0.79
Object/Concept	$\mathrm{FV}_{\mathrm{c-a}} - \mathrm{DV}_{\mathrm{c-a}} + \mathrm{DV}_{\mathrm{o-c}}$	0.1236	0.37	0.76	0.78
Object/Concept	$\mathrm{FV}_{\mathrm{c-a}} - \mathrm{DV}_{\mathrm{c-a}}$	0.1103	0.34	0.75	0.78