

Understanding Primacy Effects in Large Language Models with Sparse Autoencoders

Ikhyun Cho & Julia Hockenmaier

Siebel School of Computing and Data Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801, USA
{ihcho2, juliahmr}@illinois.edu

Abstract

The *primacy effect* is a well-established cognitive phenomenon in which information presented early in a sequence is often weighted more strongly than information presented later. Recent work suggests that large language models (LLMs) exhibit analogous behavior across a range of settings, yet the representation-level mechanisms underlying this bias remain underexplored. In this work, we provide a representation-level account of primacy effects in LLMs using sparse autoencoders (SAEs). We propose the *Dedicated Primacy Subspace* (DPS) hypothesis: LLMs appear to preferentially recruit a particular representational subspace to encode the semantics of the *first* demonstration, which subsequently modulates the processing of later context. Using few-shot classification as our testbed, we show that this mechanism has two complementary forms: *DPS-Affirmative* features, which encode support for the first-demonstration label, and *DPS-Contrastive* features, which activate when subsequent inputs contradict that initial label, thereby emphasizing competing alternatives. We further show that a subset of these features exhibits primacy-driven causal effects under intervention, providing direct causal support for the proposed mechanism. Taken together, these results provide a concrete representation-level account of how primacy effects manifest in LLMs.¹

1 Introduction

Large Language Models (LLMs) have become remarkably capable, yet they remain vulnerable to important forms of brittleness. In this paper, we study one such phenomenon: the *primacy effect*, namely the tendency for information presented early in a sequence to exert a disproportionate influence on subsequent behavior. In cognitive science, primacy refers to the tendency for early items to be remembered, weighted, or attended to more strongly than later ones (Asch, 1946; Murdock Jr, 1968). Recent work suggests that LLMs exhibit analogous behavior. For example, in multiple-choice settings, models often favor the first answer option (Zheng et al., 2024; Raimondi & Gabbrielli, 2025), while related order-sensitive biases have been observed in prompt-based classification and decision-making (Guo & Vosoughi, 2025; Guan et al., 2025; Liu et al., 2024). These effects are especially consequential for in-context learning (ICL) (Brown et al., 2020), where not only the content of demonstrations, but also their positions in the prompt, can materially shape model behavior.

Most existing work studies primacy effects at the behavioral level and seeks to mitigate them through prompt-level interventions such as prompt engineering, reranking, and calibration (Raimondi & Gabbrielli, 2025; Cho et al., 2024; Zhao et al., 2021; Guo & Vosoughi, 2025). Although such approaches are often useful in practice, they provide limited insight into *why* primacy effects arise inside the model. More broadly, recent work has argued that *order independence*—the property that a model should ideally produce the same answer under semantically equivalent reorderings of demonstrations, answer choices, or other prompt

¹Code available at <https://github.com/ihcho2/Primacy-Effects-COLM>.

elements—is desirable for reliable reasoning systems (Guan et al., 2025; Lu et al., 2022; Pezeshkpour & Hruschka, 2024). Yet whether such invariance is reflected, or even attainable, at the level of internal representations remains poorly understood.

In this work, we study primacy in the ICL setting from a mechanistic, representation-level perspective. Using sparse autoencoders (SAEs) (Huben et al., 2024; Bricken et al., 2023; Lieberum et al., 2024; Gao et al., 2025; Cho & Hockenmaier, 2025b; Cho et al., 2025b), we propose the *Dedicated Primacy Subspace* (DPS) hypothesis: LLMs preferentially use a particular subset of representation-space directions to encode the semantics associated with the *first* demonstration. This initial representation subsequently influences how later demonstrations and the final test query are represented, thereby shaping the model’s eventual prediction. Under this view, changing demonstration order does more than alter the surface arrangement of otherwise identical information: it changes which semantics are bound to these primacy-sensitive directions. Consequently, complete order invariance may be difficult to achieve at the representational level even when the semantic content of the prompt is held fixed.

We provide both distributional and causal evidence for the DPS hypothesis across multiple classification tasks (AGNews, TREC, Yahoo, and STS-B) and model families (Gemma2-2B-IT, Gemma2-9B-IT, Llama3.1-8B-IT, and Qwen3-8B-IT). Our analyses reveal two complementary families of primacy-sensitive SAE features. *DPS-Affirmative* features preferentially activate when the semantics of a later input—including the final query—are consistent with those associated with the first demonstration, thereby reinforcing the first-demonstration interpretation. In contrast, *DPS-Contrastive* features become more active when subsequent inputs are inconsistent with that interpretation, emphasizing competing semantic alternatives. Importantly, interventions on DPS features produce corresponding *first-demonstration-dependent causal effects* on the model’s output: the same fixed direction in activation space can exert systematically different, and predictable, causal effects depending on the identity of the first demonstration.

Taken together, our work makes two main contributions. First, we identify a dedicated family of primacy-sensitive representation directions, providing a concrete mechanistic account of how the first demonstration can exert persistent influence over later representations and final predictions. Second, we show that this mechanism is not monolithic: primacy is mediated by at least two complementary representational modes—an *affirmative* mode that reinforces semantic agreement with the first demonstration and a *contrastive* mode that signals semantic disagreement. Together, these findings move the study of primacy beyond behavioral-level effects and toward a mechanistic account of how prompt order is encoded and propagated within LLM representations.

2 Related Work

2.1 Primacy Effects in LLMs

A growing body of work shows that LLMs are highly sensitive to the ordering and structural placement of prompt elements. Early studies of in-context learning (ICL) showed that simply permuting the order of few-shot demonstrations can dramatically change accuracy, with some permutations yielding near-random performance and others yielding near state-of-the-art results (Lu et al., 2022). Subsequent work made these sensitivities more explicit in terms of primacy and serial-position biases. In prompt-based classification, Wang et al. (2023b) show that ChatGPT is sensitive to label order and disproportionately selects labels that appear earlier in the prompt. In multiple-choice question answering, Pezeshkpour & Hruschka (2024) show that reordering answer options can produce substantial performance shifts, indicating that option position alone can materially influence model predictions. Broadening this picture, Guo & Vosoughi (2025) document serial-position effects—including both primacy and recency biases—across a range of tasks and model families. Finally, Cobbina & Zhou (2025) show that not only the order of demonstrations but also their structural placement within the prompt can affect model outputs: relocating the same demonstration block across different prompt regions changes both accuracy and prediction stability, with early placements often yielding more stable behavior. Taken together, these

studies suggest that primacy and related order effects are not isolated artifacts, but recurring properties of LLM behavior across tasks and prompting regimes.

2.2 Behavior-Level Mitigation Approaches

Given the practical importance of these effects, much of the existing literature has focused on *behavior-level* mitigation strategies. An early and influential example is contextual calibration, which shows that normalizing model outputs using content-free inputs can substantially reduce prompt-induced bias and instability in few-shot prompting (Zhao et al., 2021). Lu et al. (2022) propose selecting effective demonstration orders by constructing an artificial development set and ranking candidate permutations using entropy-based statistics. Building on this line of work, Guo et al. (2024) study what makes a good example order in ICL and propose DEMO, an adaptive method for identifying effective example orders for individual test instances without extra in-domain data. Complementing these order-selection methods, Zhuo et al. (2024) introduce PROSA, a framework for systematically measuring prompt sensitivity, and show that few-shot examples themselves can partially alleviate prompt brittleness.

Mitigation has also been pursued through prompt redesign in more task-specific settings. Cho et al. (2024) show that reformulating exemplar answers in a comparative format, together with related prompt-structuring changes, can improve in-context classification performance. Raimondi & Gabbrielli (2025) show that primacy can sometimes be *exploited* rather than suppressed: by reordering answer options according to their semantic similarity to the query, they improve performance in multiple-choice question answering. Taken together, these studies establish a substantial literature on calibration, prompt construction, and order selection or optimization.

However, despite this empirical and engineering progress, the dominant perspective remains behavioral. Prior work typically characterizes primacy through changes in output distributions, accuracy under reorderings, label preferences, or robustness to prompt perturbations (Wang et al., 2023b; Pezeshkpour & Hruschka, 2024; Guo & Vosoughi, 2025; Zhuo et al., 2024). Even when mitigation is successful, these approaches generally do not explain *what changes inside the model* when early prompt elements exert disproportionate influence. Our work differs precisely in this respect: we use sparse autoencoders to investigate how primacy is instantiated in internal representation space and to identify the specific representational directions through which the first demonstration shapes later computation.

2.3 Sparse Autoencoders

Understanding the internal mechanisms of LLMs has long been a central goal of mechanistic interpretability (MI). Recent work in MI has advanced the *linear representation hypothesis*, which suggests that semantic concepts are often encoded as approximately linear directions in the hidden representation space of LLMs (Jiang et al., 2024; Li et al., 2023). Building on this idea, together with the *superposition hypothesis*—which proposes that LLMs represent more concepts than their hidden dimensionality allows, causing multiple concepts to become entangled within individual neurons—researchers have developed Sparse Autoencoders (SAEs). By learning an overcomplete dictionary of directions, SAEs have been shown to recover sparse and semantically meaningful directions (also called features) from LLM hidden states (Sharkey et al., 2022; Templeton et al., 2026; Cho & Hockenmaier, 2025a).

A standard SAE consists of an encoder and a decoder. Each column of the decoder matrix ($W_{dec} \in \mathbb{R}^{d \times s}$) defines an SAE feature in $\mathbb{R}^d$, where d denotes the dimensionality of the LLM hidden state. Given a hidden state, the encoder produces a sparse activation pattern over these features. The exact level of sparsity depends on the training setup, but the SAEs used in this work typically activate around a hundred features on average per token.

In general, SAEs are trained with a reconstruction objective together with a sparsity regularizer:

$$L_{SAE} = \frac{1}{N} \sum_{i=1}^N \left\| \mathbf{x}^i - \hat{\mathbf{x}}^i \right\|_2^2 + \lambda \cdot L_{\text{sparsity}}, \quad (1)$$

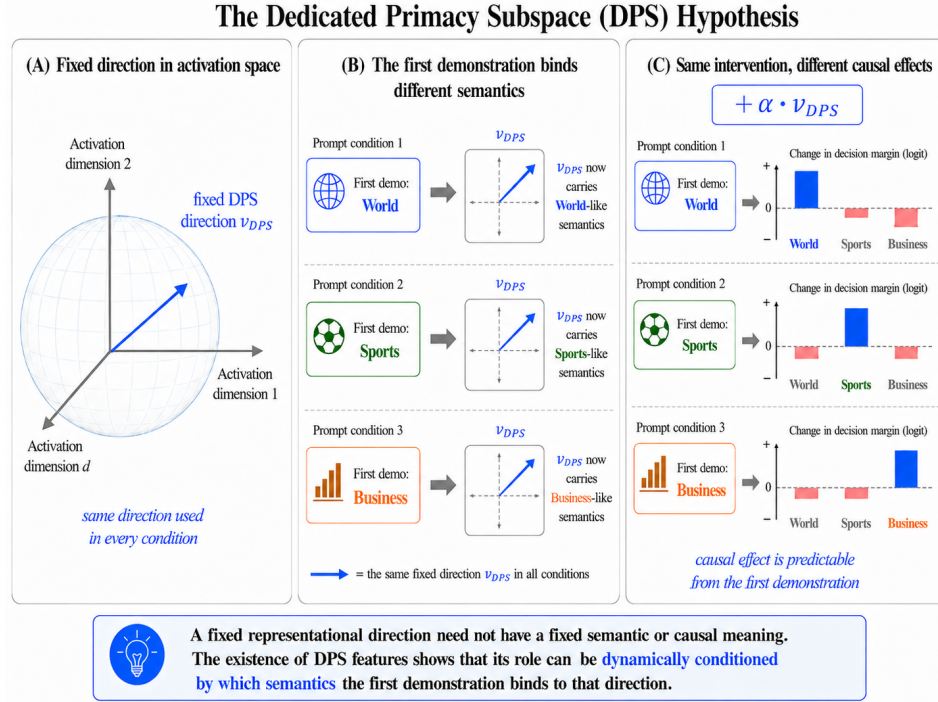


Figure 1: **Overview of the Dedicated Primacy Subspace (DPS) Hypothesis.** We hypothesize that LLMs contain fixed representational directions whose semantic and causal roles are strongly conditioned by the first demonstration. The semantics of the first demonstration become bound to these DPS features, after which the same directions are reused to process subsequent inputs, with their causal effects adapting accordingly. DPS features therefore provide a representation-level account of how primacy can be established and propagated within LLMs.

where $\mathbf{x}^i$ is the original input and $\hat{\mathbf{x}}^i$ is the SAE reconstruction:

$$\hat{\mathbf{x}}^i = W_{dec}(\sigma(W_{enc}(\mathbf{x}^i))). \quad (2)$$

Different choices of activation function have been used for σ , including JumpReLU (e.g., Gemma Scope (Lieberum et al., 2024)) and TopK-ReLU (e.g., GPT-4 SAE (Gao et al., 2025)) and Llama Scope (He et al., 2024)).

In this study, we leverage SAEs as the primary analytical tool for both formulating and evaluating the proposed *Dedicated Primacy Subspace* (DPS) hypothesis.

3 Dedicated Primacy Subspace (DPS) Hypothesis

3.1 Overview

Our central claim is that LLMs preferentially recruit a particular representational subspace to encode the semantic content of the *first* demonstration, and that sparse autoencoder (SAE) decompositions are expressive enough to recover the corresponding directions as distinct features. Crucially, this initial encoding is not confined to the representation of the first demonstration itself. Instead, it propagates forward through the prompt, shaping the representations of subsequent demonstrations and ultimately exerting influence on the model’s final prediction.

We further show that this mechanism manifests in two complementary feature families: *DPS-Affirmative* (DPS-A) and *DPS-Contrastive* (DPS-C). DPS-A features encode evidence *aligned with* the semantics of the first demonstration, whereas DPS-C features encode evidence that

a later input is *inconsistent with* those same semantics. For example, if the first demonstration is labeled SPORTS, a DPS-A feature may capture a “this looks like SPORTS” signal, whereas a DPS-C feature may capture a “this looks unlike SPORTS” signal, thereby representing evidence *against* the first-demonstration label. This distinction is mechanistically natural: effective classification requires not only accumulating evidence *for* a candidate label, but also accumulating evidence *against* competing alternatives. Our findings suggest that the model maintains both types of signals in dedicated feature families that are conditioned by the first demonstration and subsequently reused when processing later demonstrations and the final test query. Importantly, the same DPS features recur across multiple, semantically distinct tasks, suggesting that *LLMs reuse a shared set of primacy-sensitive representational directions* across tasks.

In our empirical evaluation, we analyze four instruction-tuned large language models—Gemma2-2B-IT, Gemma2-9B-IT, Llama3.1-8B-Instruct, and Qwen3-8B-IT—alongside their publicly released SAEs (Lieberum et al., 2024; He et al., 2024; Karvonen, 2025). We assess these models across three well-established text classification datasets: AGNews, which categorizes news articles into four distinct topics using their headlines and summaries (Zhang et al., 2015); TREC, a benchmark focused on sorting queries into broad semantic categories (Li & Roth, 2002); and Yahoo Answers, a ten-way topic classification task derived from user-submitted questions and their corresponding best answers (Zhang et al., 2015). Our primary analysis utilizes the representations extracted from the middle layer of each evaluated model. In Appendix D, we extend our analysis from discrete-label classification to the *continuous score-prediction* setting of STS-B and observe highly similar behavior.

3.2 Hypothesis Definition

Dedicated Primacy Subspace (DPS) Hypothesis. *There exists a largely predefined subset of representation-space directions that the model preferentially recruits to process the semantic content related to the first demonstration. This primacy-sensitive subspace is then reused throughout the remainder of the prompt, affecting the representations of later demonstrations and the final query, and thereby causally influencing the model’s final decision.*

Figure 1 provides a visual overview of the hypothesis.

3.3 Verifying the DPS Hypothesis

Why Sparse Autoencoders? Verifying the DPS hypothesis is challenging because it requires identifying a functionally meaningful subset of directions within a very high-dimensional representation space, $\mathbb{R}^d$. This is difficult for at least two reasons. First, d is large (e.g., $d = 3584$ for Gemma2-9B-IT), so locating a specific subspace associated with first-demonstration semantics is itself a nontrivial search problem. Second, individual neurons are known to be highly *polysemantic*, often responding to multiple unrelated concepts rather than cleanly encoding a single interpretable variable (Bricken et al., 2023; Elhage et al., 2022). As a result, direct neuron-level analysis is often ill-suited for detecting the subtle, context-dependent representational changes we study here.

To address this challenge, we leverage SAEs, an emerging analysis tool in the mechanistic interpretability literature. SAEs provide an efficient way to detect such changes because they decompose dense hidden states into sparse feature activations, allowing context-dependent shifts to be measured directly through *changes in feature activity*. This also offers an important advantage over methods such as probing: probing can recover directions that are predictive of a property often without guaranteeing that those directions are actually used by the model in its own computation (Belinkov, 2022; Hanna et al., 2023). By contrast, SAE features are trained specifically to reconstruct the model’s hidden states, making them more tightly coupled to the model’s internal representations and, in practice, more likely to correspond to actionable computational directions. For these reasons, we adopt SAEs as the representational language in which the DPS hypothesis is formulated and tested.

Our approach We provide evidence for the DPS mechanism by identifying two families of SAE features—DPS-AFFIRMATIVE (DPS-A) and DPS-CONTRASTIVE (DPS-C)—that exhibit strong context-adaptivity with respect to the first demonstration. Through rigorous statistical and causal intervention analyses, we show that these feature families emerge consistently and satisfy the distributional patterns predicted by the DPS hypothesis. Their existence provides direct evidence that the model recruits a dedicated representational subspace, shaped by the first demonstration, across various tasks.

Formal definitions of DPS-A and DPS-C features Let $\mathcal{Y}$ denote the label set for a classification task. For any two labels $\ell, \ell' \in \mathcal{Y}$, we construct a prompt collection $\mathcal{S}_{\ell \rightarrow \ell'}^N$, where the subscript indicates that the *first* demonstration has label ℓ and the *test query* has label ℓ' , and the superscript denotes an N -shot setting. By default, we set the number of demonstrations equal to the total number of classes (i.e., $N = |\mathcal{Y}|$). To effectively measure the influence of the first demonstration, these collections are strictly parallel: the i -th prompt in $\mathcal{S}_{\ell \rightarrow \ell'}^N$ is exactly identical to the i -th prompt in any other set $\mathcal{S}_{k \rightarrow \ell'}^N$ ($k \neq \ell$), except for the first demonstration.

Let $a_f(p)$ denote the activation of SAE feature f under prompt p , evaluated and averaged at the *label-word tokens and their immediately preceding tokens*, which have been identified as the critical positions for ICL classification (Wang et al., 2023a; Cho et al., 2025a). We define the mean response of feature f over a prompt set $\mathcal{S}$ as

$$A_f(\mathcal{S}) := \frac{1}{|\mathcal{S}|} \sum_{p \in \mathcal{S}} a_f(p).$$

Our search for DPS-AFFIRMATIVE (DPS-A) features is guided by the following intuition:

“An ideal primacy-encoding feature, if it exists, should track the semantics introduced by the first demonstration. Consequently, during the processing of a test query with label ℓ , the feature’s activation should peak if the first demonstration shares this label, and remain relatively suppressed if it presents a competing label.”

Motivated by this intuition, an ideal DPS-A feature should attain its maximum mean activation when the first demonstration also has label ℓ :

$$A_f(\mathcal{S}_{\ell \rightarrow \ell}^N) = \max_{k \in \mathcal{Y}} A_f(\mathcal{S}_{k \rightarrow \ell}^N), \quad \forall \ell \in \mathcal{Y} \quad (3)$$

That is, for a fixed test query of label ℓ , the matched condition $\mathcal{S}_{\ell \rightarrow \ell}^N$ should produce the strongest response among all possible first-demo labels.

For each SAE feature, we then evaluate how many of these label-level conditions it satisfies across all tasks. If there are M tasks and task m contains L_m labels, this yields a total of $\sum_{m=1}^M L_m$ distributional conditions. A feature that functions as a strong first-demonstration *affirmative* binder should satisfy nearly all of them.

We define DPS-CONTRASTIVE (DPS-C) features analogously, but with the reverse condition:

$$A_f(\mathcal{S}_{\ell \rightarrow \ell}^N) = \min_{k \in \mathcal{Y}} A_f(\mathcal{S}_{k \rightarrow \ell}^N), \quad \forall \ell \in \mathcal{Y} \quad (4)$$

Conceptually, these features function as a mismatch signal. They encode evidence *against* the semantics of the first demonstration, acting as an internal flag that indicates: “This query does not align with the initial demonstration.” Consequently, for a fixed test query of label ℓ , these features remain suppressed when the first demonstration shares the query’s label, but activate strongly when the demonstration belongs to a conflicting class.

Operationally, we classify a feature as DPS-A or DPS-C if it satisfies all 20 conditions of Equation (3) or Equation (4), respectively, across our evaluated datasets (4 labels for AGNews, 6 for TREC, and 10 for Yahoo). Satisfying all 20 conditions means that the *same SAE feature* is repeatedly recruited as a first-demonstration-sensitive direction across multiple semantically distinct classification tasks, providing strong empirical support for our DPS hypothesis.

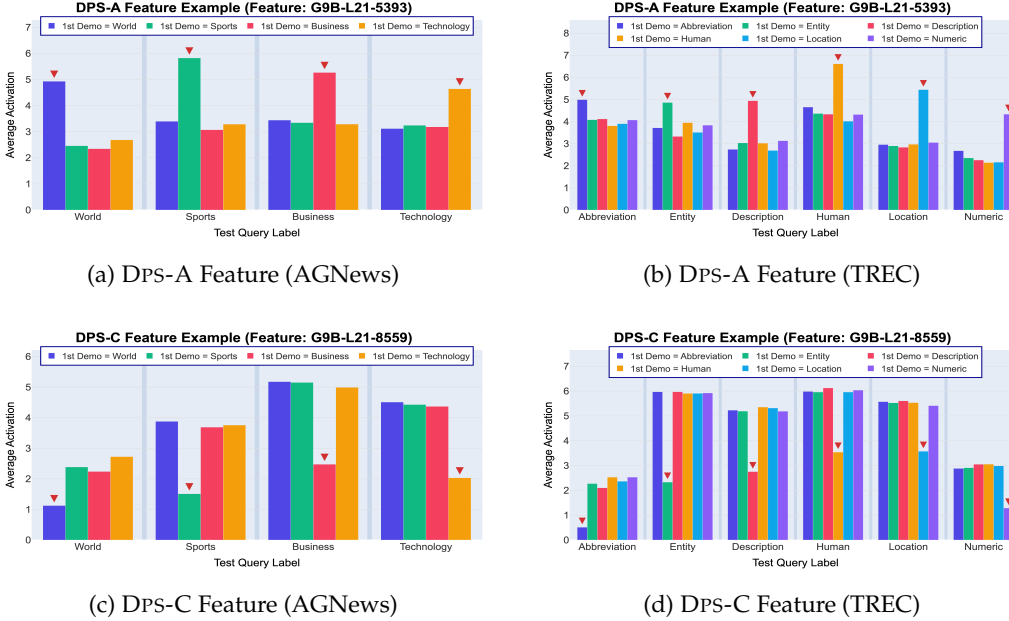


Figure 2: Examples of DPS-A (top row) and DPS-C (bottom row) features in Gemma2-9B-IT. Each sub-panel contains one group of bars for each fixed test query label $\ell \in \mathcal{Y}$; within each group, the bars show the feature’s mean activation for first-demonstration with label $\ell' \in \mathcal{Y}$. These features dynamically adapt their semantic role to the first demonstration: DPS-A features attain their maximum activation when $\ell' = \ell$, whereas DPS-C features attain their minimum activation in that same condition. The sharp contrast between the two rows suggests that primacy in LLMs is supported by at least two distinct representational mechanisms. Additional examples across other tasks and models are provided in Appendix B.

3.4 Prevalence of DPS Features

Distributional prevalence Using Equations (3) and (4), together with a minimum activation-frequency threshold of 0.01, we identify multiple features exhibiting this behavior across all evaluated models: 47 DPS-A and 50 DPS-C features for Gemma2-2B-IT, 46 and 55 respectively for Gemma2-9B-IT, and 26 and 6 for Llama3.1-8B-Instruct. Note that, in the context of SAE-based analyses, where highly consistent cross-task feature families are typically rare (Deng et al., 2025), these counts represent a nontrivial population. Taken together, the existence of these two diametrically opposed feature classes yields an important mechanistic insight: the primacy-relevant subspace is not merely a one-sided storage mechanism. Rather, it forms a richer representational system that can encode both evidence *for* and evidence *against* the semantics introduced by the first demonstration.

Figure 2 provides an illustrative example of DPS features. In the top row, a representative DPS-A feature (G9B-L21-5393) from Gemma2-9B-IT exhibits substantial shifts in its activation profile as the first-demonstration label changes: the same feature becomes maximally responsive to whichever label appears first in the prompt. This pattern recurs across multiple tasks and labels, suggesting that the model repeatedly recruits a shared representational subspace to encode first-demonstration semantics across a wide range of classification settings. In the bottom row, a representative DPS-C feature (G9B-L21-8559) exhibits the complementary pattern, showing minimal activation when the first demonstration matches the label of the test query, consistent with encoding evidence *against* that label.

3.5 The Causal Role of DPS Features

We next ask whether DPS features exert a measurable causal influence on model predictions, beyond their distributional association with first-demonstration semantics. Our main finding is that a subset of DPS features exhibits a striking form of *primacy-driven causal influence*: intervening along the same fixed SAE direction produces systematically different causal effects depending on the identity of the first demonstration.

Methodology and evaluation. For an SAE feature f at layer L , let $\mathbf{d}_{L,f} \in \mathbb{R}^d$ denote its decoder direction. For each prompt p , we intervene on the test-query token positions $T_q(p)$ by:

$$\tilde{\mathbf{h}}_{L,t}^{(f,\alpha)}(p) = \begin{cases} \mathbf{h}_{L,t}(p) + \alpha \mathbf{d}_{L,f}, & t \in T_q(p), \\ \mathbf{h}_{L,t}(p), & t \notin T_q(p), \end{cases} \quad (5)$$

where α controls the steering strength.

For a test query with gold label ℓ , let $z_\theta(c \mid p; f, \alpha)$ denote the next-token logit assigned to candidate label c after intervening on feature f with strength α . We define the correct-label decision margin as

$$m_\ell^{(f,\alpha)}(p) := z_\theta(\ell \mid p; f, \alpha) - \max_{c \in \mathcal{Y} \setminus \{\ell\}} z_\theta(c \mid p; f, \alpha). \quad (6)$$

Recall that $\mathcal{S}_{k \rightarrow \ell}^N$ denotes prompts whose first demonstration has label k and whose test query has label ℓ . We measure the causal effect of feature f under this condition by the change in the correct-label margin relative to the unsteered model:

$$\delta M_{f,\ell}(k; \alpha) := \frac{1}{|\mathcal{S}_{k \rightarrow \ell}^N|} \sum_{p \in \mathcal{S}_{k \rightarrow \ell}^N} \left[m_\ell^{(f,\alpha)}(p) - m_\ell^{(f,0)}(p) \right]. \quad (7)$$

Thus, for a fixed query label ℓ , the collection

$$\left\{ \delta M_{f,\ell}(k; \alpha) : k \in \mathcal{Y} \right\} \quad (8)$$

describes how the *same intervention* affects the model as we vary only the first demonstration.

The Ideal Causal Profile Our causal analysis is designed to test whether the effect of intervening along a fixed DPS direction is conditioned by the identity of the first demonstration. For a fixed query label ℓ , recall that $\delta M_{f,\ell}(k; \alpha)$ denotes the change in the correct-label (ℓ) margin induced by steering feature f with strength α , when the first demonstration has label k .

An ideal causal DPS-A feature should satisfy the same pattern as its distributional counterpart. Intuitively, if the feature not only activates in accordance with first-demonstration semantics but also *causally carries* those semantics, then amplifying it at the test query should have the strongest effect when the first demonstration and the query share the same label. In other words, for a fixed query label ℓ , the matched condition (i.e., $k = \ell$) should yield the largest causal effect among all possible first-demonstration labels:

$$\delta M_{f,\ell}(\ell; \alpha) = \max_{k \in \mathcal{Y}} \delta M_{f,\ell}(k; \alpha). \quad (9)$$

Analogously, an ideal causal DPS-C feature should satisfy the reverse condition:

$$\delta M_{f,\ell}(\ell; \alpha) = \min_{k \in \mathcal{Y}} \delta M_{f,\ell}(k; \alpha). \quad (10)$$

The intuition is complementary. If a DPS-C feature causally carries evidence *against* the semantics introduced by the first demonstration, then amplifying it should have the most detrimental effect when the first-demonstration label actually matches the query label. In that case, steering the feature most strongly suppresses the correct-label margin.

As in the distributional analysis, we obtain one causal-specificity condition per task label. Across AGNews, TREC, and Yahoo, this yields $4 + 6 + 10 = 20$ conditions for each feature. We evaluate the top-20 activation-level DPS-A features individually and measure how many of these 20 conditions are satisfied.

This formulation directly tests the central prediction of the DPS hypothesis: if the first demonstration binds its semantics to a DPS direction, and if that direction is genuinely used causally by the model, then intervening along the *same fixed representational direction* should produce different causal effects depending on the identity of the first demonstration.

Empirical Findings Latent-space steering is highly sensitive to intervention choices, including steering magnitude, intervention span, and the evaluation metric used to quantify behavioral change (Zou et al., 2023; Arad et al., 2025). Moreover, phenomena such as inherent self-correction in LLMs can partially obscure a feature’s causal role under standard intervention protocols (Rushing & Nanda, 2024). We therefore treat our steering setup as a conservative baseline. Despite these challenges, we identify a concrete subset of DPS-A and DPS-C candidates that satisfy most of the causal conditions defined in Equations (9) and (10). Notably, for each model, we identify at least one feature that satisfies all 20 causal conditions. This indicates that, for certain features, the first demonstration shapes not only their activation patterns but also the downstream behavioral consequences of intervening on them, providing direct causal support for the DPS hypothesis. Figure 3 illustrates representative examples of how the causal roles of DPS features adapt to the semantics established by the first demonstration.

Closing Remarks Although the causal signatures of DPS-A and DPS-C are not perfectly clean, this is consistent with the known sensitivity and noise of latent-space interventions. The central positive result is therefore not perfect causal consistency, but *existence*: across all four models, we identify concrete features whose causal roles are strongly modulated by the identity of the first demonstration (Figure 3; additional examples from other models are provided in Figure 8). These results provide feature-level evidence that primacy is not merely reflected in internal activation patterns, but can also determine the behavioral consequences of intervening along the corresponding representational directions. In Appendix E, we show that DPS features are substantially more sensitive to the first demonstration than to later demonstration positions, confirming that their behavior is specifically driven by primacy.

4 Implications of the DPS Hypothesis

Why does the DPS hypothesis matter? We highlight three main implications.

(1) A finer-grained, representation-level account of primacy effects. To the best of our knowledge, most prior work on primacy effects in LLMs has characterized the phenomenon mainly at the *behavioral level*—for example, through output biases, performance changes under prompt reordering, or prompt-engineering interventions. By contrast, our work provides a more fine-grained, *representation-level* account. We show that primacy is not simply an artifact of output behavior, but is realized in internal feature space through a compact set of directions that are preferentially engaged by the first demonstration. In this way, our work shifts the question from *whether* primacy exists to *how* it is mechanistically implemented within the model.

More specifically, we identify at least two distinct representational routes through which primacy operates in LLMs: an *affirmative* route, which encodes evidence in favor of the first-demonstration label, and a *contrastive* route, which encodes evidence that disfavors that same label. To our knowledge, this level of mechanistic specificity is new, and we believe it provides a useful foundation for future work on primacy and in-context learning.

(2) An inherent obstacle to fully eliminating primacy without representation-level intervention. Our findings also suggest that primacy effects may be fundamentally difficult to remove through prompt-level methods alone. If the model systematically reuses a largely

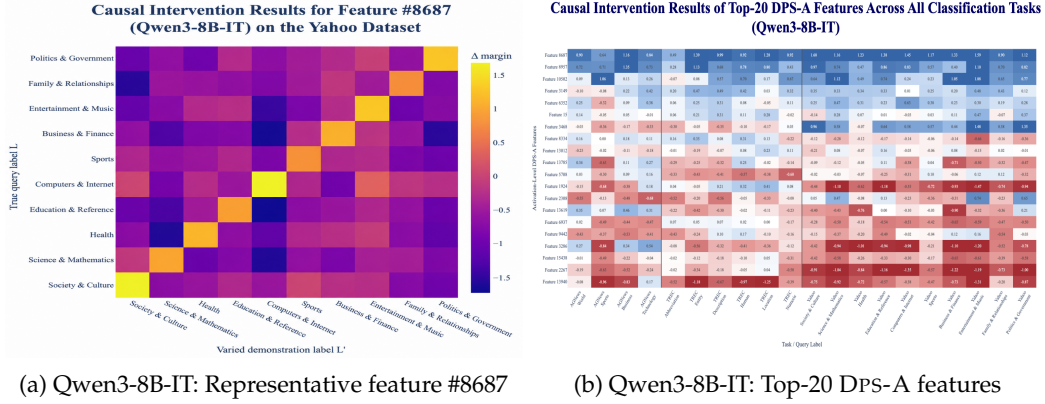


Figure 3: **Causal-DPS features exist.** (a) Full causal-effect matrix (each cell reports $\delta M_{f,\ell}(k;\alpha)$) for representative feature #8687 from Qwen3-8B-IT on Yahoo. The pronounced diagonal structure shows that the same fixed intervention direction produces systematically different effects depending on the first demonstration, providing causal support for the DPS hypothesis. (b) Causal specificity of the top-20 activation-level DPS-A features. Each cell reports $\delta M_{f,\ell}(\ell;\alpha) - \max_{k \neq \ell} \delta M_{f,\ell}(k;\alpha)$, with positive values (blue) indicating that the intervention has its strongest effect when the first-demonstration and query labels match. Two features (#8687 and #8956) satisfy all 20 causal conditions in Equation 9, while several others satisfy nearly all of them. Results on additional models are provided in Appendix C.

fixed set of feature directions to encode the semantics of the first demonstration, then changing the order of demonstrations necessarily changes which semantics are bound to those directions. In this sense, some degree of order dependence is built into the model’s internal computation, rather than being only a superficial prompt-format artifact. This does not imply that mitigation is impossible, but it does suggest that fully suppressing primacy effects may require interventions at the representation level, rather than relying exclusively on prompt engineering, reranking, or calibration.

(3) An important caveat for SAE-based interpretability research. Finally, our results have a broader methodological implication for SAE-based research. SAE features are often treated as relatively static concept detectors, and steering them is likewise expected to produce consistent behavioral effects. DPS features show that this picture is incomplete: at least some features are *context-adaptive*, with semantic roles and causal effects that change with prompt context—in our case, the identity of the first demonstration. SAE analyses should therefore avoid assuming that every feature admits a single, context-invariant interpretation. More broadly, understanding LLM internals may require distinguishing *static* from *context-adaptive* features, both of which can play important causal roles in model computation.

5 Conclusion

This paper investigates primacy effects in large language models (LLMs) from a representation-level perspective. Leveraging sparse autoencoders (SAEs), a recent advance in mechanistic interpretability, we propose the *Dedicated Primacy Subspace* (DPS) hypothesis, which posits that LLMs preferentially recruit a compact representational subspace to encode semantics associated with the first demonstration. Through both distributional and causal evidence, we provide support for this hypothesis and show that the phenomenon is shared across multiple models and tasks, suggesting that it reflects a broader and potentially generalizable mechanism. Overall, our findings offer a new representation-level account of primacy effects in LLMs and further illustrate how SAEs can serve as an effective tool for advancing our understanding of internal model computation.

References

- Dana Arad, Aaron Mueller, and Yonatan Belinkov. SAEs are good for steering – if you select the right features. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 10241–10259. Association for Computational Linguistics, November 2025. doi: 10.18653/v1/2025.emnlp-main.519. URL <https://aclanthology.org/2025.emnlp-main.519/>.
- Solomon E. Asch. Forming impressions of personality. *Journal of Abnormal Psychology*, 41: 258–90, 1946. URL <https://api.semanticscholar.org/CorpusID:28570225>.
- Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.
- Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2(5): 6, 2023.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.
- Hakaze Cho, Mariko Kato, Yoshihiro Sakai, and Naoya Inoue. Revisiting in-context learning inference circuit in large language models. In *International Conference on Learning Representations*, volume 2025, pp. 13081–13117, 2025a.
- Ikhyun Cho and Julia Hockenmaier. Analyzing multilingualism in large language models with sparse autoencoders. In *Second Conference on Language Modeling*, 2025a.
- Ikhyun Cho and Julia Hockenmaier. Toward efficient sparse autoencoder-guided steering for improved in-context learning in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 28949–28961, 2025b.
- Ikhyun Cho, Gaeul Kwon, and Julia Hockenmaier. Tutor-icl: Guiding large language models for improved in-context learning performance. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 9496–9506, 2024.
- Ikhyun Cho, Gaeul Kwon, and Julia Hockenmaier. On the versatility of sparse autoencoders for in-context learning. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 19531–19538, 2025b.
- Kwesi Adu Cobbina and Tianyi Zhou. Where to show demos in your prompt: A positional bias of in-context learning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 29548–29581, 2025.
- Boyi Deng, Yunyang Wan, Yidan Zhang, Baosong Yang, and Fuli Feng. Unveiling language-specific features in large language models via sparse autoencoders. In *Annual Meeting of the Association for Computational Linguistics*, 2025. URL <https://api.semanticscholar.org/CorpusID:278394127>.
- Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *arXiv preprint arXiv:2209.10652*, 2022.
- Leo Gao, Tom Dupre la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. In *International Conference on Learning Representations*, volume 2025, pp. 26721–26754, 2025.
- Bryan Guan, Tanya Roosta, Peyman Passban, and Mehdi Rezagholizadeh. The order effect: Investigating prompt sensitivity to input order in LLMs. *arXiv preprint arXiv:2502.04134*, 2025.

- Qi Guo, Leiyu Wang, Yidong Wang, Wei Ye, and Shikun Zhang. What makes a good order of examples in in-context learning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 14892–14904, 2024.
- Xiaobo Guo and Soroush Vosoughi. Serial position effects of large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 927–953, 2025.
- Michael Hanna, Roberto Zamparelli, and David Mareček. The functional relevance of probed information: A case study. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 835–848, 2023.
- Zhengfu He, Wentao Shu, Xuyang Ge, Lingjie Chen, Junxuan Wang, Yunhua Zhou, Frances Liu, Qipeng Guo, Xuanjing Huang, Zuxuan Wu, et al. Llama scope: Extracting millions of features from Llama-3.1-8B with sparse autoencoders. *arXiv preprint arXiv:2410.20526*, 2024.
- Robert Huben, Hoagy Cunningham, Logan Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *International Conference on Learning Representations*, volume 2024, pp. 7827–7845, 2024.
- Yibo Jiang, Goutham Rajendran, Pradeep Kumar Ravikumar, Bryon Aragam, and Victor Veitch. On the origins of linear representations in large language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 21879–21911. PMLR, 21–27 Jul 2024. URL <https://proceedings.mlr.press/v235/jiang24d.html>.
- Adam Karvonen. Qwen3-8B Sparse Autoencoders. Hugging Face model repository, 2025. URL <https://huggingface.co/adamkarvonen/qwen3-8b-saes>. BatchTopK sparse autoencoders trained at multiple layers, widths, and sparsity levels.
- Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in neural information processing systems*, 36:41451–41530, 2023.
- Xin Li and Dan Roth. Learning question classifiers. In *Coling 2002: The 19th international conference on computational linguistics*, 2002.
- Tom Lieberum, Senthoran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca Dragan, Rohin Shah, and Neel Nanda. Gemma scope: Open sparse autoencoders everywhere all at once on Gemma 2. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 278–300, 2024.
- Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the association for computational linguistics*, 12:157–173, 2024.
- Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8086–8098, 2022.
- Bennet B Murdock Jr. Serial order effects in short-term memory. *Journal of Experimental Psychology*, 76(4p2):1, 1968.
- Pouya Pezeshkpour and Estevam Hruschka. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 2006–2017, 2024.
- Bianca Raimondi and Maurizio Gabbrielli. Exploiting primacy effect to improve large language models. In *Proceedings of the 15th International Conference on Recent Advances in Natural Language Processing-Natural Language Processing in the Generative AI Era*, pp. 989–997, 2025.

- Cody Rushing and Neel Nanda. Explorations of self-repair in language models. In *Proceedings of the 41st International Conference on Machine Learning, ICML'24*. JMLR.org, 2024.
- Lee Sharkey, Dan Braun, and beren. Interim research report: Taking features out of superposition with sparse autoencoders. AI Alignment Forum / LessWrong, December 2022. URL <https://www.lesswrong.com/posts/z6QQJbtpkEAX3AoJJ/interim-research-report-taking-features-out-of-superposition>. Published December 13, 2022.
- Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, et al. Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet. *arXiv preprint arXiv:2605.29358*, 2026.
- Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. Label words are anchors: An information flow perspective for understanding in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9840–9855, 2023a.
- Yiwei Wang, Yujun Cai, Muhao Chen, Yuxuan Liang, and Bryan Hooi. Primacy effect of ChatGPT. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 108–115, 2023b.
- Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28, 2015.
- Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pp. 12697–12706. Pmlr, 2021.
- Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. Large language models are not robust multiple choice selectors. In *International Conference on Learning Representations*, volume 2024, pp. 19426–19454, 2024.
- Jingming Zhuo, Songyang Zhang, Xinyu Fang, Haodong Duan, Dahua Lin, and Kai Chen. ProSA: Assessing and understanding the prompt sensitivity of llms. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 1950–1976, 2024.
- 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 Troy Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to ai transparency. *ArXiv*, abs/2310.01405, 2023. URL <https://api.semanticscholar.org/CorpusID:263605618>.

A Prompt Templates

We utilize a standard few-shot prompt template comprising a task instruction followed by several demonstrations and a final test query. The demonstrations are randomly sampled from the training set, ensuring a balanced distribution across all labels. Figure 4 provides an implementation example of this prompt construction for the AGNews dataset using Llama3.1-8B-Instruct.

B Additional Examples of DPS-A and DPS-C Features

Figures 5, 6 and 7 display DPS-A and DPS-C feature results for Gemma2-2B-IT, Llama3.1-8B-Instruct, and Gemma2-9B-IT. The high degree of similarity across these models indicates that the primacy effect is likely driven by a shared representational mechanism.

```

def build_agnews_prompt_from_indices(
    fse_indices,
    tq_index,
    include_query_label: bool = False,
):
    """
    Prompt Construction.
    """
    parts = [
        "<|begin_of_text|><|start_header_id|>system<|end_header_id|>\n\n"
        "Pretend that you are an expert in news topic classification. "
        "For each article, determine whether its topic is one of the following: "
        "World, Sports, Business, or Technology."
    ]

    for j, fse_index in enumerate(fse_indices, start=1):
        parts.append(f"\n\nExample {j}")
        parts.append(f"\n\nArticle:\n{train_data_agnews[fse_index]['text']}")
        parts.append(f"\n\nTopic:\n{agnews_labels[get_agnews_label(fse_index)]}")

    parts.append("<|eot_id|><|start_header_id|>user<|end_header_id|>")
    parts.append(f"\n\nExample {len(fse_indices) + 1}")
    parts.append(f"\n\nArticle:\n{train_data_agnews[tq_index]['text']}")
    parts.append(f"\n\nTopic:\n")

    parts.append("<|eot_id|><|start_header_id|>assistant<|end_header_id|>\n\n")

    if include_query_label:
        parts.append(agnews_labels[get_agnews_label(tq_index)] + "<|eot_id|>")

    return "".join(parts)

```

Figure 4: Example of prompt construction for the AGNews dataset using Llama3.1-8B-IT.

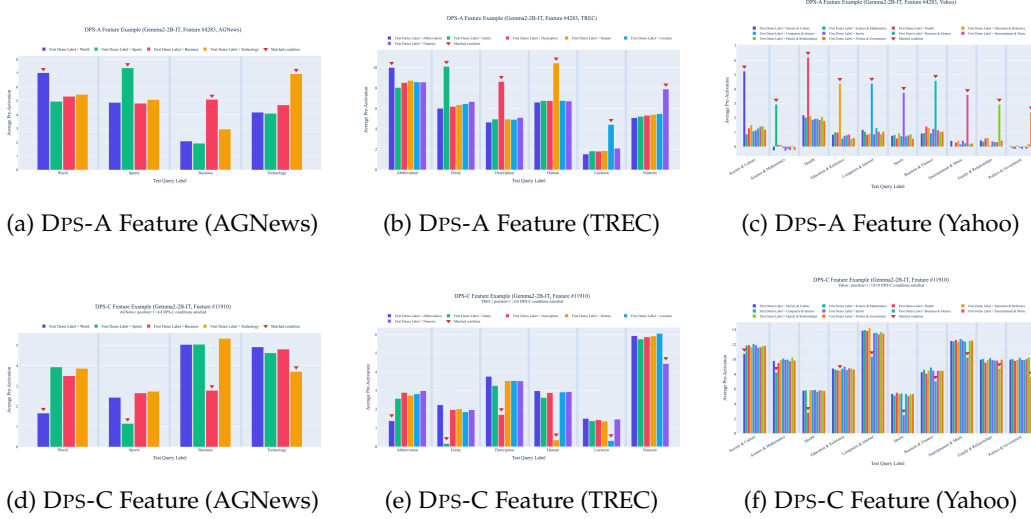


Figure 5: Additional Examples of DPS-A (top row) and DPS-C (bottom row) features in Gemma2-2B-IT.

C Extended Causal Results

Figure 8 provides additional causal evaluations of DPS-A and DPS-C features for Llama3.1-8B-Instruct, Gemma2-2B-IT, and Gemma2-9B-IT.

D Extension to the STS-B Task

Our main experiments study few-shot classification, where the semantic content associated with the first demonstration is naturally represented by a discrete class label. We additionally

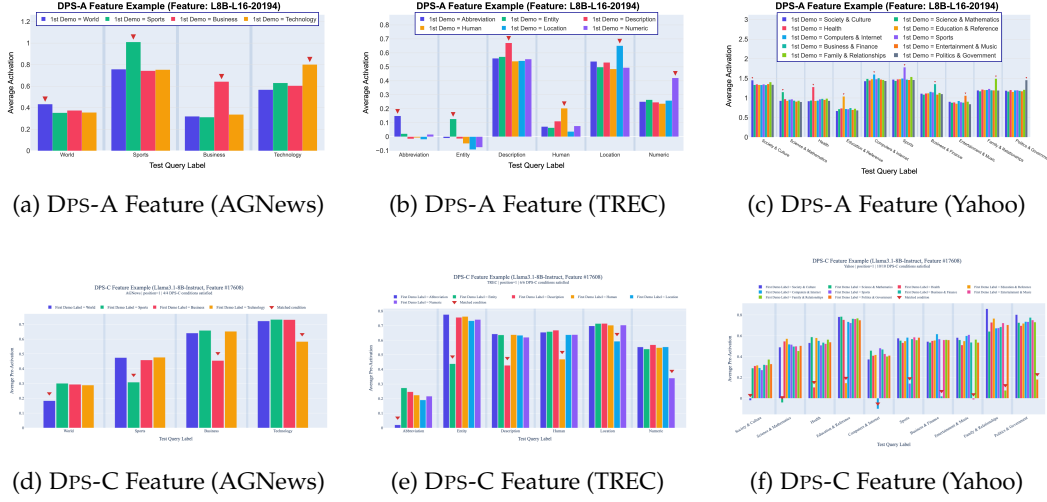


Figure 6: Additional Examples of DPS-A (top row) and DPS-C (bottom row) features in Llama3.1-8B-Instruct.

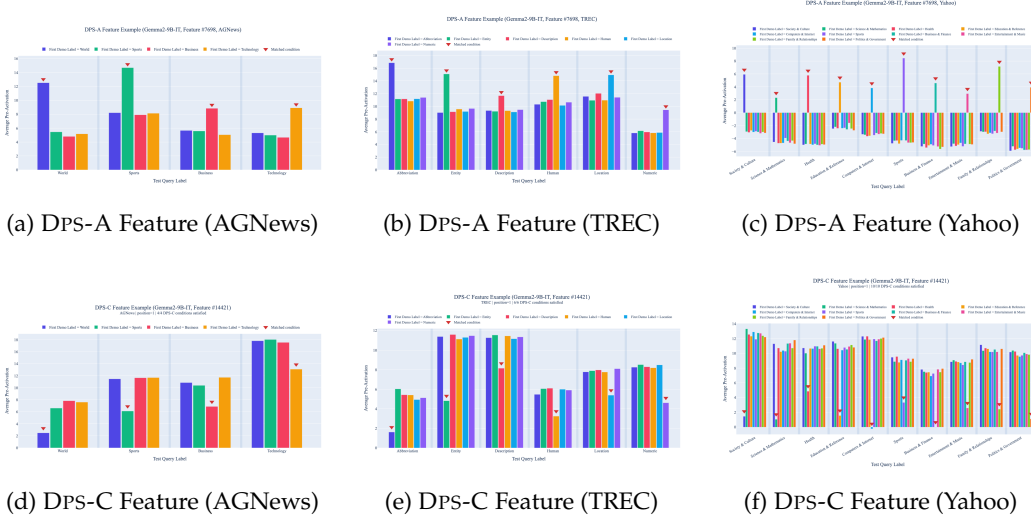


Figure 7: Additional Examples of DPS-A (top row) and DPS-C (bottom row) features in Gemma2-9B-IT.

examine whether the same phenomenon extends to a *continuous* prediction setting using STS-B, where each sentence pair is assigned a semantic-similarity score between 0 and 5. This setting provides a useful complementary test of the DPS hypothesis: rather than asking whether a feature preferentially represents one of several discrete labels, we can ask whether its causal effect continuously adapts to the scalar semantics established by the first demonstration.

Experimental setup. We construct few-shot STS-B prompts in which the first demonstration is systematically varied across score anchors

$$s_1 \in \{0, 1, 2, 3, 4, 5\},$$

while holding the test query and the remaining demonstrations fixed. We use the model’s expected score over candidate numeric outputs as its prediction. Let $\hat{y}_{\text{base}}$ denote the

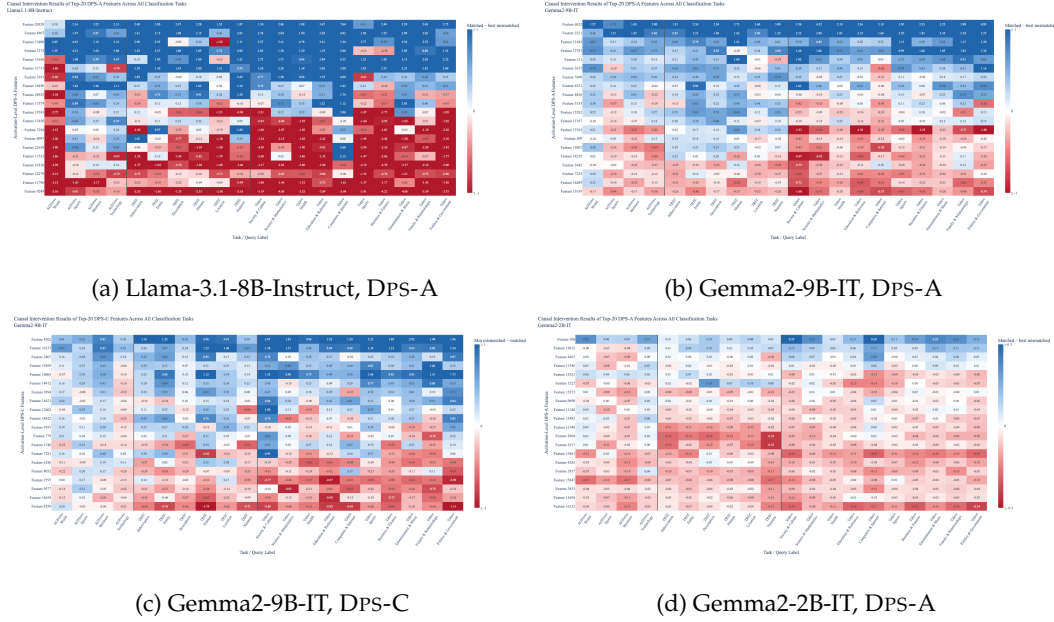


Figure 8: Across all models, we identify at least one DPS-A and one DPS-C feature that satisfies all 20 causal conditions, corresponding to an all-blue first row in each heatmap. The red cells further highlight that causal effects need not align perfectly with the corresponding distributional patterns.

model’s baseline prediction for a given prompt and $\hat{y}_{\text{steer}}$ its prediction after amplifying a DPS-A direction. We define the intervention-induced score shift as

$$\Delta\hat{y} = \hat{y}_{\text{steer}} - \hat{y}_{\text{base}}. \quad (11)$$

If the causal role of a DPS direction is conditioned by the first demonstration, then the same fixed intervention direction should not simply increase or decrease the predicted score uniformly. Instead, its effect should vary with the score semantics established by the first demonstration.

The same feature produces first-demonstration-dependent score shifts. Figure 9 illustrates this behavior for representative DPS-A feature #6629 in Qwen3-8B-IT. Each cell reports the mean score shift $\Delta\hat{y}$ for a particular combination of test-query score anchor and first-demonstration score anchor, using the *same fixed SAE direction* and intervention strength $\alpha = 100$ throughout. The intervention produces both positive and negative effects depending on the first-demonstration context. In particular, high first-demonstration scores frequently induce substantial upward shifts in the model’s prediction, whereas sufficiently low first-demonstration scores can induce downward shifts. Thus, the feature does not behave as a context-independent “increase score” or “decrease score” direction. Rather, its causal effect changes systematically with the score semantics supplied by the first demonstration.

Steering pulls predictions toward the first-demonstration score. To quantify this effect more directly, we measure whether the intervention moves the model’s prediction closer to the score of the first demonstration. For first-demonstration score s_1 , we define

$$R = |s_1 - \hat{y}_{\text{base}}| - |s_1 - \hat{y}_{\text{steer}}|. \quad (12)$$

Positive R indicates that steering reduces the distance between the model’s predicted test-query score and the first-demonstration score.

Figure 10 shows a clear positive effect across all baseline-distance bins. More importantly, the effect becomes substantially stronger when the first-demonstration score is far from the

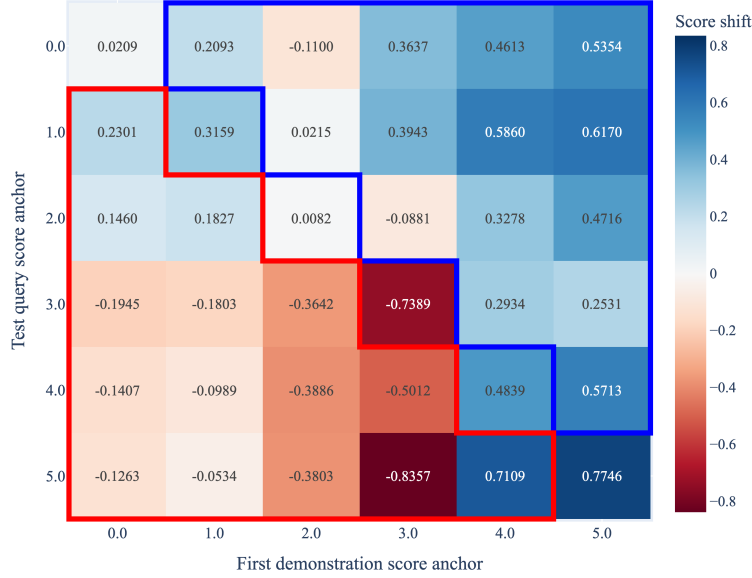
Feature 6629: mean score shift grid ($\alpha=100$)

Figure 9: **A fixed DPS direction produces first-demonstration-dependent causal effects on STS-B.** Mean intervention-induced score shift $\Delta\hat{y}$ for representative DPS-A feature #6629 in Qwen3-8B-IT with $\alpha = 100$. Rows correspond to test-query score anchors and columns to first-demonstration score anchors. Despite using the same fixed SAE direction in every condition, the magnitude and sign of the resulting score shift vary substantially with the semantics established by the first demonstration.

model’s original prediction. When $|s_1 - \hat{y}_{\text{base}}| < 3.5$, the mean reduction is comparatively modest, approximately 0.15–0.27. This increases to 0.32 for distances between 3.5 and 4, 0.69 between 4 and 4.5, and 1.10 when the first-demonstration score differs from the baseline prediction by at least 4.5 points. Hence, the causal effect is strongest precisely when the semantic value established by the first demonstration differs most strongly from the model’s otherwise preferred prediction.

Implication for the DPS hypothesis. These results provide a continuous-valued analogue of our classification findings. In classification, a fixed DPS direction exhibits different causal effects depending on the discrete semantics associated with the first demonstration. On STS-B, the same phenomenon appears along a continuous semantic dimension: steering the same fixed feature direction pulls the model’s subsequent prediction toward the scalar score established by the first demonstration, with the strength of this pull increasing as the first-demonstration score becomes more discrepant from the model’s baseline prediction. This provides further evidence that the causal role of a DPS direction is not intrinsically tied to a single fixed semantic interpretation, but is instead conditioned by the semantic content bound to it by the first demonstration.

E Position-Specificity of DPS Features

To further verify that DPS features are specifically driven by the *first* demonstration, rather than by demonstration position more generally, we examine how their characteristic activation pattern changes when the varied demonstration is moved to later positions in the prompt. We use Qwen3-8B-IT and DPS-A features as representative examples, and evaluate

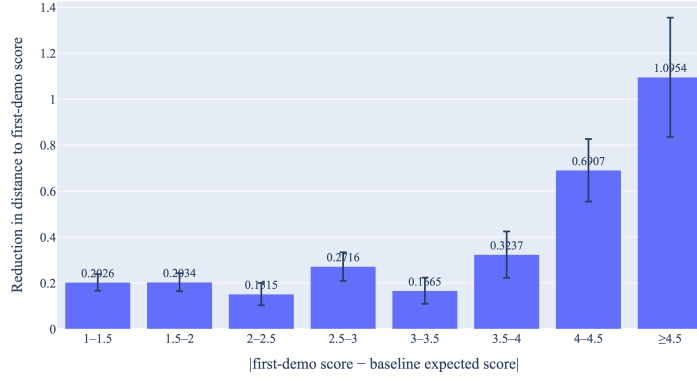
Feature 6629: distance-to-first-demo reduction ($\alpha=100$)

Figure 10: **Steering a DPS feature pulls the predicted STS-B score toward the first-demonstration score.** We report the reduction in distance to the first-demonstration score, $|s_1 - \hat{y}_{\text{base}}| - |s_1 - \hat{y}_{\text{steer}}|$, for representative DPS-A feature #6629 in Qwen3-8B-IT with $\alpha = 100$, grouped by the initial distance between the first-demonstration score and the model’s baseline prediction. Positive values indicate movement toward the first-demonstration score. The effect is positive across all bins and becomes particularly pronounced when the first-demonstration score is far from the baseline prediction.

the same three classification tasks used in our main cross-task analysis: AGNews, TREC, and Yahoo.

Experimental setup. Recall that a DPS-A feature satisfies, for each query label ℓ ,

$$A_f(\mathcal{S}_{\ell \rightarrow \ell}^N) > \max_{k \in \mathcal{Y} \setminus \{\ell\}} A_f(\mathcal{S}_{k \rightarrow \ell}^N), \quad (13)$$

i.e., its activation is largest when the varied demonstration and the test query share the same label. Across AGNews, TREC, and Yahoo, this yields $4 + 6 + 10 = 20$ label-level conditions.

We first identify DPS-A features using the standard setting in which the varied demonstration occupies the first demonstration position. We then *freeze this feature set* and evaluate the same features on held-out examples while moving the varied demonstration to positions $1, \dots, 9$. We use the first nine positions to provide a broad position range that is shared across all three task configurations.

Importantly, for a given test query, the identity of the varied demonstration, the remaining demonstrations, and the query itself are held fixed across positions; only the location of the varied demonstration changes. Thus, changes in the DPS activation pattern can be attributed to demonstration position rather than differences in prompt content.

For each feature f and demonstration position p , we count the number of cross-task DPS-A conditions that it continues to satisfy:

$$C_f(p) = \sum_{m=1}^M \sum_{\ell \in \mathcal{Y}_m} 1 \left[A_{f,m}^{(p)}(\ell \rightarrow \ell) > \max_{k \in \mathcal{Y}_m \setminus \{\ell\}} A_{f,m}^{(p)}(k \rightarrow \ell) \right], \quad C_f(p) \in \{0, \dots, 20\}, \quad (14)$$

where m indexes the three tasks. We then measure the fraction of features originally identified at position 1 that retain their complete 20-condition DPS-A profile at position p :

$$R(p) = \frac{\left| \left\{ f \in \mathcal{F}_{\text{DPS-A}}^{(1)} : C_f(p) = 20 \right\} \right|}{\left| \mathcal{F}_{\text{DPS-A}}^{(1)} \right|}, \quad (15)$$

where $\mathcal{F}_{\text{DPS-A}}^{(1)}$ denotes the DPS-A features identified from the first-demonstration condition.

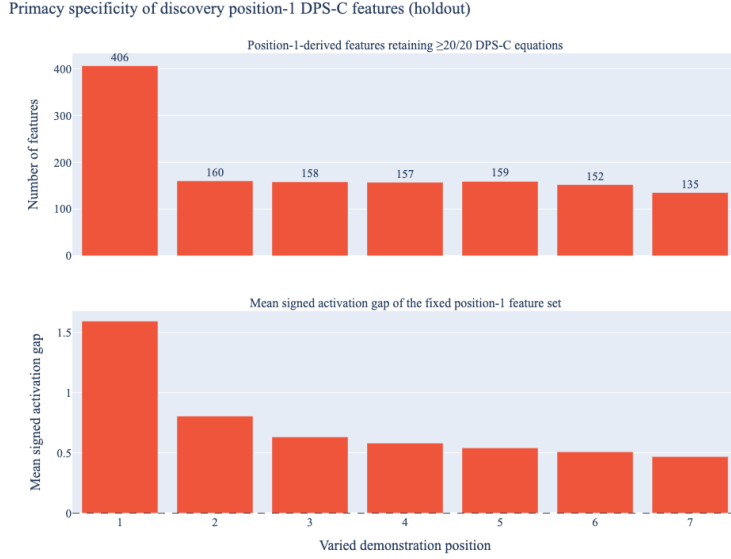


Figure 11: **DPS-A features are strongly specific to the first demonstration position.** We identify DPS-A features at position 1 and freeze this feature set before evaluating it on held-out examples as the varied demonstration is moved to later positions. *Top:* the number of position-1-derived features that continue to satisfy all 20 cross-task DPS-A conditions drops sharply beyond the first position. *Bottom:* the mean signed activation gap of the same fixed feature set likewise decreases substantially with later demonstration positions. Together, these results indicate that the DPS-A pattern is specifically associated with primacy rather than generic sensitivity to demonstration content.

Results. Figure 11 reveals a pronounced position-specific effect. Features identified as DPS-A at the first demonstration rapidly lose their defining DPS-A profile as the varied demonstration is moved to later positions. In particular, the number of position-1 DPS-A features that continue to satisfy all 20 cross-task conditions drops sharply away from position 1.

This result rules out a more generic interpretation in which these features simply respond to the semantics of whichever demonstration is manipulated. If that were the case, their characteristic activation pattern should remain largely intact when the same demonstration is moved elsewhere in the prompt. Instead, the effect is strongly privileged at the *first* demonstration position. Thus, the DPS feature family is not merely demonstration-sensitive, but specifically *primacy-sensitive*, providing additional evidence that the mechanism identified in the main text reflects a genuine internal primacy effect.