Where is “Washing Machine” Stored in LLMs? Compound Concepts Are Compositionally Derived, Not Holistically Represented

Anonymous Author(s)

Abstract

Large language models must represent millions of compound concepts—“washing machine,” “hot dog,” “coffee table”—yet their residual streams have limited dimensionality. How do these multi-token concepts fit? We investigate whether compound nouns are stored as unique directions in the residual stream or are compositionally derived from their constituent words. Through four complementary experiments on GPT-2 (124M parameters), we find that compounds are primarily represented through composition: a linear combination of constituent word directions explains 93.7% of the variance in compound representations ($R^2 = 0.937$), and the dominant mechanism is next-token prediction boosting—seeing “washing” raises $P($“machine”$)$ by a median factor of $20\times$ compared to control contexts. A linear probe achieves 92.2% accuracy at distinguishing compound from non-compound contexts, confirming that compounds carry contextual information beyond their parts, but this signal accounts for only $\sim 6\%$ of the total representation. We find no evidence of token erasure in GPT-2: constituent word identity is perfectly recoverable from the compound position at every layer. More idiomatic compounds (e.g., “hot dog,” $R^2 = 0.888$) require more unique representational capacity than transparent ones (e.g., “steel bridge,” $R^2 = 0.965$). Our findings challenge the assumption that multi-token concepts have dedicated directions and have direct implications for concept editing, representation steering, and sparse autoencoder-based interpretability.

1 Introduction

How does a language model know that “washing machine” refers to a household appliance rather than a machine that happens to be washing? The linear representation hypothesis [Park et al., 2024] suggests that concepts correspond to directions in a model’s activation space, but this has only been tested for single-token concepts. Multi-token compound nouns—“washing machine,” “hot dog,” “swimming pool”—pose a fundamental representational challenge: there are far more compound concepts in language than dimensions in a model’s residual stream. GPT-2 has 768 dimensions; even the superposition hypothesis [Elhage et al., 2022], which allows $\sim 10\times$ more features than dimensions through near-orthogonal packing, cannot accommodate the millions of possible compounds.

How, then, are compound concepts stored? Three possibilities exist. First, the model may develop *holistic* representations—unique directions for “washing machine” assembled at the second token position. Second, the model may rely on *compositional* processing, where “washing” has a direction that makes “machine” contextually likely, with no unified compound direction. Third, a *hybrid* of both mechanisms may operate.

Prior work has not addressed this question directly. Park et al. [2024] tested 27 single-token concepts in LLAMA-2-7B but never multi-token compounds. Feucht et al. [2024] showed that named entities exhibit token erasure—information about preceding tokens is rapidly transformed at the last token

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

position—but studied proper nouns in Llama-2, not common compound nouns in smaller models. Ormerod et al. [2024] probed compound noun semantics in BERT, a masked language model, but not in autoregressive models where the sequential processing of compounds is fundamentally different. No study has directly compared compound concept directions against constituent word directions in the residual stream of autoregressive transformers.

We address this gap with four complementary experiments on GPT-2 (124M parameters, 12 layers, $d_{\text{model}} = 768$), examining 19 compound nouns that span the full compositionality spectrum from transparent ("steel bridge") to idiomatic ("hot dog"). Our experiments measure (1) next-token prediction boosting, (2) linear reconstructability of compound directions from constituents, (3) layer-wise probing for token erasure and compound-specific information, and (4) attention patterns between compound constituents. We validate key findings on GPT-2-MEDIUM (355M parameters, 24 layers).

Our results support the compositional hypothesis. The compound direction at the second token position is 93.7% reconstructable as a linear combination of constituent directions ($R^2 = 0.937 \pm 0.020$). The primary mechanism is next-token prediction boosting: seeing "washing" makes "machine" the top prediction with $P = 0.827$, a $4{,}963\times$ boost over the control word "red." We find no token erasure—constituent identity is perfectly recoverable at every layer—and the 6% of compound representations not explained by constituents is sufficient to distinguish compound from non-compound contexts (92.2% probe accuracy) but does not constitute a dedicated compound direction.

In summary, we make the following contributions:

- We conduct the first systematic analysis of how compound nouns are represented in the residual stream of autoregressive language models, testing holistic versus compositional storage across 19 compounds varying in compositionality.
- We show that compound representations are 93.7% linearly reconstructable from constituent word directions, with more idiomatic compounds requiring more unique capacity ($R^2 = 0.888$ for "hot dog" vs. 0.965 for "steel bridge"; Spearman $\rho = 0.669$, $p = 0.006$).
- We demonstrate that the dominant mechanism for compound processing is next-token prediction boosting (median $20\times$ boost), not the construction of holistic compound representations, and find no evidence of token erasure in GPT-2—contradicting the implicit vocabulary hypothesis for this model class.

2 Related Work

Linear representations in LLMs. The linear representation hypothesis posits that concepts correspond to directions in a model's activation space. Park et al. [2024] formalized three variants—subspace, measurement, and intervention—and tested 27 single-token concepts in LLAMA-2-7B, finding strong linear structure for 26 of 27. The only failure was the compositional relation "thing→part," hinting that compositional concepts may not follow the same pattern. Gurnee and Tegmark [2023] showed that LLMs develop linear representations of space and time, while Merullo et al. [2023] demonstrated that LLMs implement Word2Vec-style vector arithmetic for relational tasks. Unlike these studies, which focus on single-token concepts or simple relations, we directly test whether *multi-token* compound nouns have independent directions or are compositionally derived from their parts.

Superposition and polysemanticity. Elhage et al. [2022] established that neural networks store more features than they have dimensions by packing sparse features as nearly-orthogonal directions—the superposition hypothesis. Scherlis et al. [2022] showed that polysemanticity emerges naturally from capacity constraints. These results imply that compound concepts like "washing machine," being relatively sparse in training data, are almost certainly stored in superposition with other concepts rather than in dedicated neurons. Our work tests a complementary question: whether compound concepts have *any* dedicated direction (even in superposition) or are entirely derived from constituent directions.

Sparse autoencoders and feature discovery. Sparse autoencoders (SAEs) decompose polysemantic activations into interpretable features [Cunningham et al., 2023, Gao et al., 2024, Rajamanoharan et al., 2024]. Pre-trained SAEs are now available for multiple model families [Lieberum et al., 2024, He et al., 2024]. However, Chanin et al. [2024] identified the feature absorption problem: when features form hierarchies, SAEs absorb parent features into child features, creating unreliable

Compound	w_1	w_2	**Control**	**Comp.**
washing machine	washing	machine	red machine	4
hot dog	hot	dog	big dog	1
coffee table	coffee	table	wooden table	5
swimming pool	swimming	pool	deep pool	4
parking lot	parking	lot	large lot	4
living room	living	room	large room	4
shooting star	shooting	star	bright star	3
chocolate cake	chocolate	cake	large cake	5
steel bridge	steel	bridge	old bridge	5

Table 1: Representative compounds from our dataset with compositionality ratings (1=idiomatic, 5=transparent). Full dataset contains 19 compounds. All compounds tokenize as exactly two tokens in GPT-2's BPE vocabulary.

classifiers. Chanin et al. [2025] further showed that narrow SAEs merge correlated features. These findings suggest that SAE features for compounds may be unreliable—motivating our direct analysis of residual stream directions rather than relying on SAE decompositions.

Multi-token concept processing. Feucht et al. [2024] showed that at the last token position of multi-token entities, information about preceding tokens is rapidly erased in layers 1–9 of Llama-2-7B, suggesting an "implicit vocabulary" of multi-token items. We apply a similar probing methodology to compound nouns in GPT-2 and find a strikingly different result: no token erasure occurs, suggesting that compound nouns and named entities may be processed differently, or that model scale plays a role. Geva et al. [2022] showed that feed-forward layers build next-token predictions by promoting concepts in the vocabulary space, which is consistent with our finding that FFN layers gradually increase P("machine") after processing "washing."

Compositionality and idiomaticity. Ormerod et al. [2024] used representational similarity analysis to compare transformer representations of compound nouns with human judgments, finding that compounds processed together have different representations than constituents processed separately in BERT-family models. Our work extends this to autoregressive models and directly quantifies the compositional versus unique components of compound representations. Aljaafari et al. [2024] found that no specific layer integrates tokens into unified semantic representations based on constituent parts, with information distributed across layers—consistent with our U-shaped R^2 pattern across layers. Garcia et al. [2021] tested whether BERT captures idiomatic versus compositional meanings and found that contextualized models fail to accurately distinguish them, prioritizing lexical overlap. Dankers et al. [2022] showed that transformers tend to over-generate compositional translations of idioms, suggesting a bias toward compositional processing. Building on these findings, we provide the first quantitative decomposition of compound representations into compositional and unique components in autoregressive LLMs.

3 Methodology

We conduct four experiments on GPT-2 [Radford et al., 2019] (124M parameters, 12 layers, $d_{\text{model}} = 768$), accessing internal activations via TransformerLens [Nanda and Bloom, 2022]. We validate key findings on GPT-2-Medium (355M parameters, 24 layers, $d_{\text{model}} = 1024$).

3.1 Dataset Construction

We construct a dataset of 19 compound nouns spanning the compositionality spectrum, each paired with a control phrase that preserves the second word (w_2) but replaces the first word (w_1) with a non-compound modifier. table 1 shows representative examples. Compositionality ratings range from 1 (fully idiomatic, e.g., "hot dog") to 5 (fully transparent, e.g., "coffee table").

All compounds are verified to tokenize as exactly two tokens in GPT-2's BPE vocabulary. Two potential compounds were excluded: "blueberry" ("berry" is multi-token) and "guinea pig" from direction analysis ("guinea" is multi-token in some contexts). Each compound is embedded in 8 diverse sentence templates (e.g., "The {compound} was," "She bought a {compound} for") to ensure context diversity. For isolation baselines, each constituent word appears in 4 templates without its compound partner.

3.2 Experiment 1: Next-Token Prediction Analysis

For each compound (w_1, w_2), we measure how strongly w_1 predicts w_2 compared to a control word. Specifically, we compute:

$$\text{Boost} = \frac{P(w_2 \mid \text{context} + w_1)}{P(w_2 \mid \text{context} + w_1^{\text{ctrl}})} \tag{1}$$

where w_1^{ctrl} is the control word (e.g., "red" for "washing machine"). We average probabilities across 8 sentence templates and also record the rank of w_2 among all vocabulary predictions.

3.3 Experiment 2: Residual Stream Direction Analysis

We test whether the compound direction can be linearly reconstructed from constituent directions. At each layer ℓ, we collect:

- $\boldsymbol{h}^{(\ell)}_{\text{compound}}$: the mean hidden state at the w_2 position across 8 compound contexts.
- $\boldsymbol{h}^{(\ell)}_{w_1}$: the mean hidden state for w_1 in isolation (4 contexts).
- $\boldsymbol{h}^{(\ell)}_{w_2}$: the mean hidden state for w_2 in isolation (4 contexts).

We fit the linear reconstruction:

$$\boldsymbol{h}^{(\ell)}_{\text{compound}} = \alpha \cdot \boldsymbol{h}^{(\ell)}_{w_1} + \beta \cdot \boldsymbol{h}^{(\ell)}_{w_2} \tag{2}$$

using ordinary least squares, and report the coefficient of determination R^2. We also compute cosine similarities between compound and constituent directions, and the residual norm ratio $\|\boldsymbol{h}_{\text{compound}} - \hat{\boldsymbol{h}}_{\text{compound}}\| / \|\boldsymbol{h}_{\text{compound}}\|$, which measures the fraction of the compound representation that is unique (i.e., not explained by constituents).

3.4 Experiment 3: Layer-wise Probing

We train two linear probes (logistic regression, $C = 1.0$, 5-fold cross-validation) on hidden states at the w_2 position:

Probe 1 (Token erasure). Predicts the identity of w_1 from the hidden state at the w_2 position. If the model erases constituent information to form compound representations, probe accuracy should decrease in intermediate layers.

Probe 2 (Compound detection). Classifies whether the context is a compound (e.g., "washing machine") or a control phrase (e.g., "red machine"). This measures how much compound-specific information is present at each layer. We use 120 compound samples and 136 control samples.

3.5 Experiment 4: Attention Pattern Analysis

We extract attention weights at the w_2 position and compare how much w_2 attends to w_1 in compound contexts versus how much w_2 attends to the preceding word in control contexts. We analyze this across all layers and attention heads.

4 Results

4.1 Next-Token Prediction Boosting

The primary mechanism for compound processing is next-token prediction boosting. table 2 shows that for strongly associated compounds, w_2 is the top prediction after w_1: "washing" yields

Compound	$P(w_2 \mid w_1)$	**Rank**	$P(w_2 \mid w_1^{\text{ctrl}})$	**Ctrl Rank**	**Boost**
guinea pig	0.833	**1**	0.0001	1755	7,233×
washing machine	0.827	**1**	0.0002	924	4,963×
swimming pool	0.626	**1**	0.0011	125	549×
parking lot	0.322	**1**	0.0116	14	28×
living room	0.261	2	0.0016	98	160×
driving license	0.036	118	0.0002	1325	221×
hot dog	0.097	4	0.0022	142	45×
coffee table	0.063	4	0.0091	14	7×
chocolate cake	0.036	5	0.0003	978	140×
snowman	0.004	63	0.043	7	0.1×
sunflower	0.0001	1673	0.0004	706	0.3×

Table 2: Next-token prediction results for selected compounds. The boost ratio (equation 1) measures how much more likely w_2 is after w_1 compared to a control word. Strong compounds show boost ratios of 28–7,233×, while single-token compounds (snowman, sunflower) show ratios below 1.

P(“machine”) $= 0.827$ (rank 1), “guinea” yields P(“pig”) $= 0.833$ (rank 1), and “swimming” yields P(“pool”) $= 0.626$ (rank 1). The median boost ratio across all compounds is $20.2\times$ (95% CI: $[4.7, 180.2]$), which is statistically significant (Wilcoxon signed-rank $W = 160$, $p = 2.1 \times 10^{-4}$, Cohen’s $d = 0.63$).

Two compounds—snowman and sunflower—show boost ratios below 1, meaning the control word actually predicts w_2 better. This is expected: these compounds are written as single words in standard English (“snowman,” “sunflower”), so the model’s training data does not contain “snow” and “man” as adjacent tokens in compound contexts.

4.2 Compound Directions Are 93.7% Reconstructable

table 3 presents the linear reconstruction results at the final layer. The mean R^2 across all 15 tested compounds is 0.937 ± 0.020 ($t = 176.3$, $p = 7.9 \times 10^{-25}$ vs. the null of $R^2 = 0$). The mean cosine similarity between compound and w_2 directions is 0.959, and between compound and w_1 directions is 0.926. The mean residual norm ratio is 0.253, indicating that only $\sim$25% of the compound representation’s norm is not explained by constituents.

Compositionality predicts reconstruction quality. There is a significant positive correlation between compositionality rating and R^2 (Spearman $\rho = 0.669$, $p = 0.006$): transparent compounds like “steel bridge” ($R^2 = 0.965$) are better reconstructed than idiomatic ones like “hot dog” ($R^2 = 0.888$). This is exactly what the compositional hypothesis predicts—idiomatic compounds require more unique information beyond their constituents.

U-shaped R^2 across layers. R^2 starts high at layer 0 (0.940), drops to a minimum at layers 4–5 ($\sim$0.800), and recovers to 0.937 at layer 11. This suggests that intermediate layers perform the most compound-specific processing, potentially encoding compound-specific semantics, before later layers restore more compositional representations for next-token prediction.

4.3 No Token Erasure in GPT-2

Probe 1 results (token erasure). The token identity probe achieves *perfect accuracy* (1.000) at every layer from 0 to 11. The identity of w_1 is fully recoverable from the hidden state at the w_2 position throughout the network. This stands in contrast to Feucht et al. [2024], who found that token identity accuracy drops from $\sim$100% at layer 0 to $\sim$20% by layer 9 in LLAMA-2-7B for named entities.

Probe 2 results (compound detection). The compound-vs-control probe shows that compound-specific information emerges early and peaks in later layers (table 4). Accuracy is 70.6% at layer 0, rises above 90% by layer 2, and peaks at 92.2% at layer 8. The slight decrease to 85.1% at layer

Compound	R^2	$\cos(\mathbf{c}, w_1)$	$\cos(\mathbf{c}, w_2)$	**Residual**	**Comp.**
steel bridge	**0.965**	0.924	0.982	0.188	5
garden hose	0.962	0.933	0.979	0.196	5
door handle	0.958	0.949	0.976	0.206	5
mountain cabin	0.956	0.937	0.975	0.209	5
chocolate cake	0.953	0.944	0.975	0.216	5
coffee table	0.938	0.934	0.962	0.249	5
swimming pool	0.924	0.932	0.957	0.276	4
washing machine	0.917	0.907	0.951	0.288	4
hot dog	0.888	0.897	0.934	0.335	1
Mean	0.937	0.926	0.959	0.253	—

Table 3: Linear reconstruction of compound directions from constituent directions at the final layer. R^2 measures the fraction of variance explained by the linear model in equation 2. Residual is the normalized reconstruction error. Comp. is the compositionality rating (1=idiomatic, 5=transparent). More compositional compounds have higher R^2.

Layer	0	2	4	7	8	11
Token erasure (Probe 1)	1.000	1.000	1.000	1.000	1.000	1.000
Compound detection (Probe 2)	0.706	0.902	0.894	0.918	**0.922**	0.851

Table 4: Probing accuracy at selected layers. Probe 1 tests whether w_1 identity can be recovered from the w_2 position (perfect accuracy = no erasure). Probe 2 tests whether compound versus control context can be distinguished. Peak compound detection accuracy is **92.2%** at layer 8.

11 may reflect the model optimizing late-layer representations for next-token prediction rather than maintaining compound-specific information.

4.4 Attention Patterns

Compound contexts show significantly more attention from w_2 to w_1 in layer 0 ($p = 0.011$), but this pattern reverses in later layers (7–8), where control contexts show more attention from w_2 to the preceding word ($p < 0.001$). This suggests that the model uses early-layer attention to establish the compound relationship, then shifts to different processing in later layers once the compound context has been encoded into the residual stream.

4.5 Validation on GPT-2-MEDIUM

Key findings replicate on GPT-2-MEDIUM (355M parameters, 24 layers). table 5 shows that the larger model produces similar or stronger next-token prediction boosting and comparable R^2 values. The pattern of compositional storage is robust to a $2.9\times$ increase in model size.

5 Discussion

5.1 Compound Concepts Are Compositionally Derived

Our four experiments converge on a consistent picture: compound nouns in GPT-2 are not stored as unique directions in the residual stream. Instead, the model uses two complementary mechanisms. First, *next-token prediction boosting*: the representation of w_1 (“washing”) causes the model to assign high probability to w_2 (“machine”) as the next token. Second, *contextual modulation*: the hidden state at the w_2 position is a linear combination of constituent representations ($R^2 = 0.937$) with a small unique component ($\sim 6\%$) that carries compound-specific information.

Compound	Boost		R^2	
	GPT-2	GPT-2-MEDIUM	GPT-2	GPT-2-MEDIUM
washing machine	4,963×	25,303×	0.917	0.899
swimming pool	549×	615×	0.924	0.932
hot dog	45×	116×	0.888	0.897
coffee table	7×	4×	0.938	0.923

Table 5: Validation on GPT-2-MEDIUM. Next-token prediction boost ratios and R^2 reconstruction scores are comparable or stronger than GPT-2, confirming that compositional storage is robust to model scale.

This finding is consistent with Aljaafari et al. [2024], who found no single layer integrating tokens into unified semantic representations, and with Dankers et al. [2022], who observed a bias toward compositional processing in transformers. However, our quantitative decomposition goes further: we show that the compositional component accounts for the vast majority (94%) of the representation, with the unique component being small but detectable.

5.2 The Role of Compositionality

The correlation between linguistic compositionality and R^2 (Spearman $\rho = 0.669, p = 0.006$) reveals that the model allocates more unique representational capacity to semantically opaque compounds. "Hot dog" ($R^2 = 0.888$, compositionality = 1) has the most unique content of any tested compound, while "steel bridge" ($R^2 = 0.965$, compositionality = 5) is almost entirely derived from its parts. This suggests that the model learns to devote additional capacity—the $\sim 11\%$ unique component for "hot dog" versus $\sim 4\%$ for "steel bridge"—to encoding the non-compositional semantics that cannot be derived from the meanings of "hot" and "dog" separately.

An intriguing exception is "guinea pig" (compositionality = 2), which despite being semantically opaque has the highest next-token boost ratio (7,233×). This dissociation between semantic compositionality and statistical predictability confirms that the model's next-token boosting mechanism relies on co-occurrence statistics rather than semantic understanding.

5.3 No Token Erasure: A Model-Scale Effect?

Our finding of perfect token identity recovery at every layer in GPT-2 contradicts Feucht et al. [2024], who observed strong erasure in LLAMA-2-7B. Several explanations are possible. First, model scale: GPT-2 has 124M parameters versus 7B for Llama-2, and smaller models may lack the capacity to develop implicit vocabulary representations. Second, entity type: Feucht et al. [2024] studied named entities ("Space Needle"), which may be processed differently from common compound nouns ("washing machine"). Third, architecture differences between GPT-2 and Llama-2 (rotary position embeddings, different normalization) could affect how multi-token information is propagated. Resolving this requires testing compound nouns specifically in larger models.

5.4 The U-Shaped R^2 Pattern

The dip in R^2 at layers 4–5 (~ 0.800) followed by recovery at layer 11 (0.937) suggests a two-phase processing pipeline. In early-to-mid layers, the model performs compound-specific transformations that move the representation away from a simple constituent combination—potentially encoding semantic relationships or pragmatic associations. In later layers, the representation is restructured toward a form optimized for next-token prediction, which apparently resembles the compositional combination more closely. This is consistent with Geva et al. [2022]'s finding that FFN layers build predictions by promoting concepts in vocabulary space: the final layers may "undo" some of the compound-specific processing to produce a representation that correctly predicts the next token.

5.5 Implications

For mechanistic interpretability. Compound concepts challenge the "one concept = one direction" view. If "washing machine" does not have its own direction, SAE features for compounds will likely be unreliable—consistent with the feature absorption problem [Chanin et al., 2024]. Interpretability methods should account for the fact that multi-token concept representations are distributed across constituent directions plus contextual modulation.

For concept editing. Editing "washing machine" by modifying a single direction would likely fail. Instead, one would need to modify the contextual relationship between "washing" and "machine" representations, which involves attention patterns and FFN transformations distributed across the network. This has practical implications for knowledge editing and model debiasing approaches that assume concepts have localized linear representations.

For the linear representation hypothesis. The hypothesis holds approximately for compounds: compound representations are linear combinations of constituent representations. However, the $\sim 6\%$ unique component and the U-shaped layer evolution suggest that the full picture involves nonlinear dynamics in intermediate layers. The linear representation hypothesis, as tested by Park et al. [2024], should be extended to explicitly consider multi-token concepts.

5.6 Limitations

Model scale. GPT-2 is small by modern standards. Larger models may develop more holistic compound representations, particularly given the token erasure findings in LLAMA-2-7B [Feucht et al., 2024]. Our validation on GPT-2-MEDIUM shows similar patterns, but testing on 7B+ models is needed.

Limited context diversity. We use 8 sentence templates per compound. Natural corpus contexts from large-scale text (e.g., The Pile) would better capture the range of compound usage and reduce template bias.

English only. All compounds are English. Languages with productive compounding (e.g., German, where novel compounds are formed freely) may reveal different patterns.

Linear probing. Linear probes may miss nonlinear compound representations. MLP probes or representation similarity analysis could capture additional structure.

Control phrase selection. The specific choice of control words (e.g., "red" for "washing machine") affects boost ratios. A larger set of control words would provide more robust estimates.

Small dataset. With 19 compounds, our statistical power is limited. The correlation between compositionality and R^2 ($p = 0.006$) is significant, but a larger dataset (100+ compounds with validated compositionality ratings) would strengthen this finding.

6 Conclusion

We investigated how compound nouns are represented in the residual stream of autoregressive language models. Through four experiments on GPT-2 (validated on GPT-2-MEDIUM), we find that compound concepts like "washing machine" are not stored as unique directions. Instead, the model uses compositional mechanisms: the compound direction is 93.7% reconstructable from constituent word directions, and the dominant processing mechanism is next-token prediction boosting (median 20× boost). We observe no token erasure in GPT-2—constituent identity is perfectly recoverable at every layer—and the small unique component ($\sim 6\%$) of compound representations, while sufficient to distinguish compound from non-compound contexts (92.2% probe accuracy), does not constitute a dedicated compound direction.

These findings have direct implications for mechanistic interpretability, concept editing, and the linear representation hypothesis. Multi-token compound concepts are best understood not as points in activation space but as contextual modulations of constituent representations. Future work should extend this analysis to larger models where token erasure has been observed, use natural corpus contexts, and test whether the compositional storage pattern holds across languages and for longer multi-word expressions.

References

Nura Aljaafari, Yonatan Belinkov, and Fazl Barez. Interpreting token compositionality in LLMs: A robustness analysis. *arXiv preprint arXiv:2410.12924*, 2024.

David Chanin, Fazl Barez, Adamk Karvonen, and Neel Nanda. A is for absorption: Studying feature splitting and absorption in sparse autoencoders. *arXiv preprint arXiv:2409.14507*, 2024.

David Chanin, Fazl Barez, and Adam Karvonen. Feature hedging: On the role of feature redundancy in sparse autoencoders. *arXiv preprint arXiv:2503.01370*, 2025.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*, 2023.

Verna Dankers, Christopher G Lucas, and Ivan Titov. Can transformer be too compositional? analysing idiom processing in neural machine translation. *arXiv preprint arXiv:2205.15301*, 2022.

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.

Sheridan Feucht, David Torné, and David Bau. Token erasure as a footprint of implicit vocabulary items in LLMs. *arXiv preprint arXiv:2406.20086*, 2024.

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093*, 2024.

Marcos Garcia, Tiago Kramer Vieira, Carolina Scarton, Marco Idiart, and Aline Villavicencio. Probing for idiomaticity in vector space models. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, 2021.

Mor Geva, Avi Caciularu, Kevin Ro Wang, and Yoav Goldberg. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. *arXiv preprint arXiv:2203.14680*, 2022.

Wes Gurnee and Max Tegmark. Language models represent space and time. *arXiv preprint arXiv:2310.02207*, 2023.

Zhengfu He, Wentao Ge, Muhao Chen, and Zhuowan Liu. Llama scope: Extracting millions of features from Llama-3.1-8B with sparse autoencoders. *arXiv preprint arXiv:2410.20526*, 2024.

Tom Lieberum, Senthooran 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. *arXiv preprint arXiv:2408.05147*, 2024.

Jack Merullo, Carsten Eickhoff, and Ellie Pavlick. Language models implement simple word2vec-style vector arithmetic. *arXiv preprint arXiv:2305.16130*, 2023.

Neel Nanda and Joseph Bloom. TransformerLens. 2022. `https://github.com/neelnanda-io/TransformerLens`.

Mark Ormerod, Jesús Maldonado, and Roman Klinger. How is a “kitchen chair” like a “farm horse”? exploring the representation of compound nominals in transformer-based language models. *Computational Linguistics*, 50(1), 2024.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2024.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 2019.

Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Tom Lieberum, Vikrant Varma, János Kramár, Rohin Shah, and Neel Nanda. Improving dictionary learning with gated sparse autoencoders. *arXiv preprint arXiv:2404.16014*, 2024.

Adam Scherlis, Kshitij Sachan, Adam S Jermyn, Joe Benton, and Buck Shlegeris. Polysemanticity and capacity in neural networks. *arXiv preprint arXiv:2210.01892*, 2022.