Sparse Auto-Encoder Interprets Linguistic Features in Large Language Models

Anonymous ACL submission

Abstract

Large language models (LLMs) excel in tasks that require complex linguistic abilities, such as reference disambiguation and metaphor recognition/generation. Although LLMs possess impressive capabilities, their internal mechanisms for processing and representing linguistic knowledge remain largely opaque. Previous work on linguistic mechanisms has been limited by coarse granularity, insufficient causal analysis, and a narrow focus. In this study, we present a systematic and comprehensive causal investigation using sparse auto-encoders (SAEs). We extract a wide range of linguistic features from six dimensions: phonetics, phonology, morphology, syntax, semantics, and pragmatics. We extract, evaluate, and intervene on these features by constructing minimal contrast datasets and counterfactual sentence datasets. We introduce two indices—Feature Representation Confidence (FRC) and Feature Intervention Confidence (FIC)—to measure the ability of linguistic features to capture and control linguistic phenomena. Our results reveal inherent representations of linguistic knowledge in LLMs and demonstrate the potential for controlling model outputs. This work provides strong evidence that LLMs possess genuine linguistic knowledge and lays the foundation for more interpretable and controllable language modeling in future research.

1 Introduction

Large language models (LLMs) exhibit excellent performance in solving tasks that require different levels of linguistic competence, such as dependency parsing (Lin et al., 2022; Roy et al., 2023), reference disambiguation (Iyer et al., 2023) and metaphor interpretation (Wachowiak and Gromann, 2023; Yerukola et al., 2024; Tian et al., 2024). While their linguistic capabilities are largely attributed to the emergence of abilities from large-scale pre-training and model size (Manning et al.,

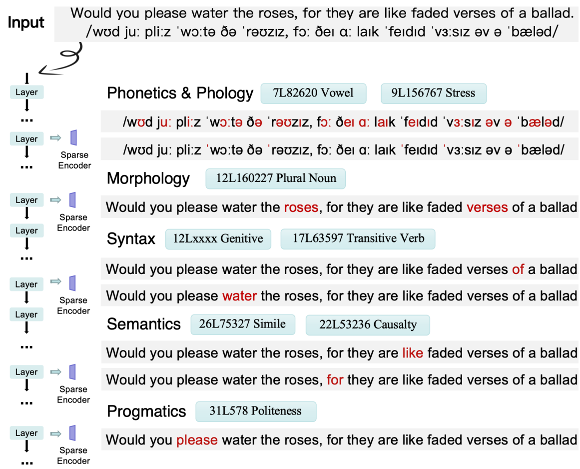

Figure 1: After a sentence is input into the model, its hidden states are encoded by the SAE into a sparse feature distribution. Across multiple layers, we can identify base vectors that are significantly activated and associated with the sentence's linguistic features.

2020; Allen-Zhu and Li, 2023; Mahowald et al., 2024), the underlying mechanisms by which LLMs process these linguistic structures remain underexplored (Saba, 2023). Thus, we aim to interpret the linguistic mechanisms of LLMs by addressing the following question: *Can we identify minimal components within LLMs that are responsible for distinct linguistic processing capabilities?*

Previous attempts at interpreting the linguistic mechanisms of LLMs usually involve instructing them with expert-designed prompts, aiming to explain how these models generate particular outputs. (Yin and Neubig, 2022). Nevertheless, such behavior-based methods do not provide model-structure-level mechanism interpretation. Most recent works turn to establish the connection between specific linguistic capabilities of LLMs and their interior structure, such as hidden states (Katz and Belinkov, 2023), attention heads (Wu et al., 2020), and activated neurons (Sajjad et al., 2022; Huang et al., 2023). However, these approaches mostly suffer from the following challenges:

Coarse Interpretation Granularity. Linguistic mechanism interpretation aims to find the *atomic* linguistic structure in LLMs. However, even neurons are the most fine-grained native components of LLMs, they are observed to be activated by multiple different conditions, a phenomena that are termed as poly-semanticity (Yan et al., 2024). Thus, it is necessary to extract more fine-grained structures from LLMs to interpret their linguistic mechanism.

Insufficient Causal Analysis. Current linguistic mechanism interpretations successfully identify relevant inner structure of LLMs, with activation patching (Hanna et al., 2024; Nanda et al., 2023) as the most typical methodology. However, it is still challenging to verify causal relationships between linguistic abilities and corresponding internal structures, which is a prerequisite for effectively steering model behavior through interventions on corresponding mechanisms.

To address these challenges, we propose to utilize sparse auto-encoder (**SAE**) for interpreting **ling**uistic mechanisms of LLMs, a framework dubbed SAELING. SAE learns a projection matrix which decomposes the hidden states of LLMs into an extremely high-dimensional feature space with sparse activation constraint, where each dimension is expected to represent a single meaning, as Figure 1 shows. Building on this, SAELING comprises two components: (1) SAELING designs sparse feature analysis for un-interpreted features that are extracted by SAE, which provides a fine-grained linguistic mechanism interpretation for LLMs; (2) SAELING proposes to manipulate the LLMs via intervening on the features with desired interpretation, which verifies the causal relationship between the features and their interpretations. It also potentially paves way to steer the linguistic behavior of LLMs.

In particular, we first establishes a hierarchical linguistic framework with annotated corpora. The framework classifies the linguistic features into six categories, including phonetics, phonology, morphology, syntax, semantics, and pragmatics. These linguistic features are wildly observed linguistic abilities, thus guarantee the feasibility to interpret their mechanisms. To interpret sparse features in SAEs, we construct minimal pairs and counterfactual sentences for each sentence in our dataset. We also introduce a causal analysis method that intervenes on specific linguistic features via the SAE and uses an LLM as a judge to assess the intervention effect. Furthermore, we present two causality evaluation metrics: the Feature Representation Confidence (FRC) score and Feature Intervention Confidence (FIC) score, which measure a feature's ability to identify the corresponding linguistic phenomenon in the input and its ability to regulate the model output to generate the phenomenon, respectively.

We conduct a series of experiments on Llama-3.1-8B (Grattafiori et al., 2024). Our experiment results show that SAELING effectively identifies key features for linguistic competence. SAELING also provides a robust way to steer LLMs by intervening on the found linguistic features.

2 Related Works

Linguistic mechanism interpretation has been a ever-chasing goal since the emergence of LLMs. We review linguistic capability evaluation for LLMs and corresponding mechanistic interpretation works. We will also introduce the basic concepts for sparse auto-encoder.

Linguistic Features in LLMs. LLMs are shown to be equipped with diverse linguistic features. Morphological studies find inflectional and derivational phenomena along with word-formation processes in LLMs (Rambelli et al., 2024; Weissweiler et al., 2023). Syntactic evaluations include canonical constructions, *e.g.,* genitives and object-complement structures (Gauthier et al., 2020; Zhang et al., 2023; Arora et al., 2024), and cross-linguistic tests (Mueller et al., 2020). Semantic investigations address metaphor comprehension (Tian et al., 2024; Wachowiak and Gromann, 2023; Stowe et al., 2021; He et al., 2022; Liu et al., 2022), deep semantic analysis (Chen et al., 2024), and output consistency (Raj et al., 2023). Pragmatic benchmarks examine the interpretation of contextual cues (Sileo et al., 2022; Wu et al., 2024).

Linguistic Mechanism Interpretation. LLMs excel in most of the above tasks, which spurs growing interest in explaining their linguistic capabilities. At the behavioral explanation level, methods include feature attribution, contrastive explanation (Yin and Neubig, 2022), surrogate model explanation, and self-explanation. At the hidden-layer explanation level, approaches comprise analyses of attention heads (Wu et al., 2020), probing tasks (Hahn and Baroni, 2019; Arora et al., 2024; He et al., 2022), and correlation studies (Liu et al.,

2024) of hidden-layer activation patterns. At the neuron explanation level, research has primarily focused on analyzing the activations of linguistically relevant neurons (Tang et al., 2024).

Sparse Auto-encoder. Recent work has employed sparse auto-encoders (SAEs) to interpret the hidden-layer activations of large language models by decomposing them into a large set of concept features (Gao et al., 2024). These concept features exhibit mono-semanticity and hold considerable interpretability potential (Huben et al., 2024). In particular, an SAE maps the hidden states $\mathbf{f} \in \mathbb{R}^d$ in LLMs into the feature space with sparse activations:

$$\mathbf{f} = \text{SparseConstraint}\left(\mathbf{W}_e \mathbf{h} + \mathbf{b}_e\right),$$

where the SAE is parameterized by $\mathbf{W}_e \in \mathbb{R}^{(r\times d)\times d}$, $\mathbf{b}_e \in \mathbb{R}^{(r\times d)}$. r is the expansion ratio, defined as the factor by which the hidden state dimension is expanded. Commonly used sparse constraint include TopK (Gao et al., 2024) and JupeReLU (Rajamanoharan et al., 2024) functions. As each dimension of the sparse activation in $\mathbf{f}$ corresponds to a base vector in $\mathbf{W}_e$, this paper uses base vector to denote features extracted by SAE.

3 Methodology

SAELING consists of three key components. (1) A hierarchical linguistic structure with supporting corpora for linguistic mechanism analysis; (2) Linguistic feature analysis for interpreting SAE extracted features; and (3) Linguistic feature intervention for causal analysis and LLM steering.

3.1 Linguistic Structure

Hierarchical Linguistic Structure. To systematically interpret the language capabilities of large models, we adopt a six level structure based on theoretical linguistics (Fromkin et al., 2017): phonetics, phonology, morphology, syntax, semantics, and pragmatics. The structure follows a logical progression from the external, physical realization of sound to the internal, contextual understanding of meaning. Each linguistic capability contains several concrete linguistic features, *e.g.,* semantics level includes metaphor, simile, *etc*. We provide the exact definition for each linguistic capability in Appendix A.1

Dataset Construction. The sparse feature activation distribution of SAE is closely related to the conditions under which their corresponding linguistic features hold in linguistic knowledge. To find the linguistic features and evaluate its dominance, we propose a method to construct the dataset and analyze feature activation frequencies.

For each linguistic feature, we first construct a set of sentences that significantly align with the desired feature. The feature activation representing this linguistic feature in SAE's hidden space will be significantly activated on these sentences. However, this is not enough to accurately identify them, as there are some background noise vectors that are activated on all sentences in the dataset and interfere with our judgment. We need to include a control group without the feature in the constructed sentences.

We introduce two types of control groups: minimal pairs and counterfactual sentences. Minimal pairs are constructed by changing only the part of a sentence that corresponds to a particular linguistic feature, while keeping all other parts unchanged. However, this approach often results in syntactically incorrect sentences.

To overcome this limitation, we also construct fully grammatically correct control groups, called counterfactual sentences, which differ from the original sentence only in terms of its linguistic features. Detailed dataset construction procedures are provided in Appendix A.2.

3.2 Feature Analysis

We propose a causal probability approach to evaluate the relationship between extracted linguistic features and their activation on sentences containing those features.

For a given feature x, we define two key probabilities. The *Probability of Necessity* (PN) quantifies how necessary the feature is for the activation of a corresponding base vector, while the *Probability of Sufficiency* (PS) measures the likelihood that introducing the feature triggers activation. These probabilities are then combined into a *Feature Representation Confidence* (FRC) score, which assesses both the representational capacity of the SAE latent space and the discriminative ability of the feature to identify the corresponding linguistic phenomenon.

During feature analysis, we calculate the FRS on both the minimal contrast dataset and the counterfactual dataset, then average the results. This average more accurately reflects the ability of the base vectors to represent the linguistic features. Detailed definitions and calculation methods are

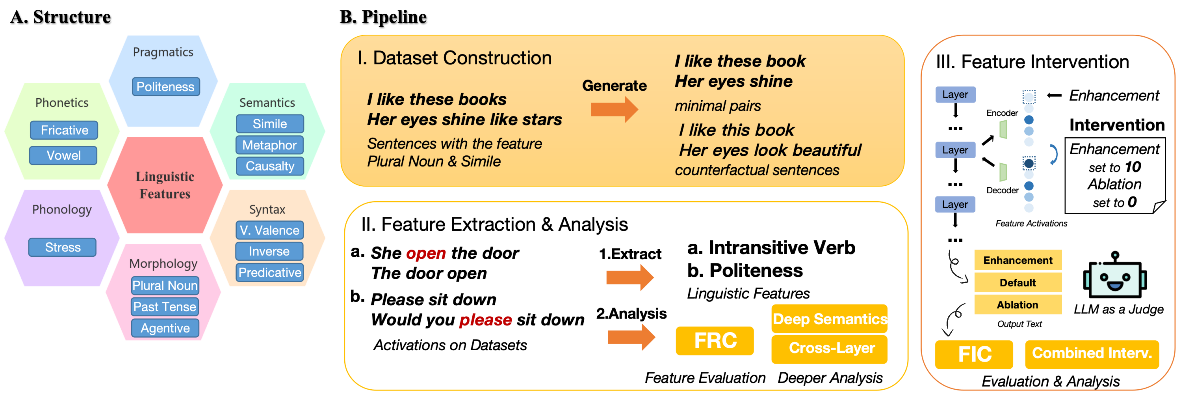

Figure 2: The overall framework of SAELING. We propose a large-model linguistic mechanism framework encompassing six dimensions and select classical features from these dimensions for experimentation. The experimental workflow is as follows: (1) Construct minimal contrast and counterfactual datasets; (2) Extract features and evaluate their relevance by analyzing the activation values of base vectors on the datasets; (3) Intervene in the model output by modifying activation values and assess causality using an LLM as a judge.

provided in Appendix D.1.

3.3 Feature Intervention

When we modify the values of SAE's activation during forward propagation, we expect that such targeted interventions will influence the model's behavior. However, our experiments show that altering only a small subset of features may not significantly impact the output—likely because linguistic phenomena are represented by multiple features across various layers. To assess the true impact of these interventions, we use a large language model as a judge. For each linguistic feature, we conduct both ablation and enhancement experiments. In the ablation experiment, we set the target feature's activation to 0, and in the enhancement experiment, we set it to 10. In both cases, we also perform baseline experiments by randomly selecting 25 base vectors from the same layer.

For brevity, we denote the interventions as follows: let I^T_{abl} denote the targeted ablation intervention, I^B_{abl} the baseline ablation intervention, I^T_{enh} the targeted enhancement intervention, and I^B_{enh} the baseline enhancement intervention.

Let P^T_{abl} and P^B_{abl} denote the success probabilities (*i.e.*, the probability that the intended change in the linguistic phenomenon is observed) for the targeted and baseline ablation experiments. The normalized ablation effect is then defined as

$$\begin{aligned} E_{abl} &= P^T_{abl} - P^B_{abl} \\ &= \frac{P(Y=0 \mid I^T_{abl}) - P(Y=0 \mid I^B_{abl})}{P(Y=0 \mid I^T_{abl})}. \end{aligned}$$

Similarly, let P^T_{enh} and P^B_{enh} be the success probabilities for the targeted and baseline enhancement experiments, with $Y = 1$ indicating the presence of the phenomenon. The normalized enhancement effect is given by

$$\begin{aligned} E_{enh} &= P^T_{enh} - P^B_{enh} \\ &= \frac{P(Y=1 \mid I^T_{enh}) - P(Y=1 \mid I^B_{enh})}{1 - P(Y=1 \mid I^B_{enh})}. \end{aligned}$$

Finally, we define the Feature Intervention Confidence (FIC) score as the harmonic mean of the normalized ablation and enhancement effects:

$$\text{FIC} = \frac{2\, E_{abl}\, E_{enh}}{E_{abl} + E_{enh}}.$$

When calculating FIC, if one or both of the E values are negative, we incorporate a penalty coefficient w to reflect the weakened or lost causality in such cases. This FIC score provides a balanced measure of how effectively targeted interventions, as opposed to random ones, influence the model's output with respect to specific linguistic features. The details for FIC are shown in Appendix D.2.

4 Experiments

4.1 Experiment Setup

Model. We conduct experiments on Llama-3.1-8B (Grattafiori et al., 2024). For SAEs, we use OpenSAE (THU-KEG, 2025) and its released checkpoints on 32 layers of Llama-3.1-8B.

Dataset. For feature analysis, we select 18 canonical linguistic phenomena from the renowned textbook *An Introduction to Language* (Fromkin et al., 2017) as experimental features. The selected

Feature	ID	M-Pairs		C-Fact		FRC
		PS	PN	PS	PN	
Phonetics						
Sibilant	7L230243	100.0	100.0	100.0	100.0	100.0
Vowel	7L82620	100.0	100.0	100.0	100.0	100.0
Phonology						
Stress	9L156767	100.0	100.0	100.0	100.0	100.0
Morphology						
Past-Tense	8L4016	100.0	100.0	100.0	100.0	100.0
Agentive Suffix	12L199760	100.0	95.0	100.0	92.5	96.9
Plural Noun	12L160227	100.0	97.5	100.0	100.0	99.4
Syntax						
Intransitive Verb	17L63597	/	/	92.5	97.5	94.9
Transitive Verb	17L174515	/	/	100.0	90.0	94.7
Linking Verb	18L61112	/	/	90.0	95.0	92.4
Inverse	21L17802	/	/	100.0	100.0	100.0
Genitive	20L259762	100.0	100.0	100.0	100.0	100.0
Semantics						
Causality	22L223621	100.0	100.0	100.0	100.0	100.0
Adversativity	24L7721	87.5	100.0	87.5	100.0	93.3
Progression	25L60962	100.0	100.0	100.0	100.0	100.0
Metaphor	26L253776	68.8	100.0	68.8	88.8	81.6
Simile	26L75327	95.0	96.3	86.3	91.3	92.2
Pragmatics						
Discourse Marker	27L173147	92.5	92.5	92.5	92.5	92.5
Politeness	31L578	/	/	100.0	96.3	98.1

Table 1: Feature analysis results. Probability of Necessity (PN) and Probability of Sufficiency (PS) of the extracted linguistic features (Feature, layer, ID) on Minimal-Pair and Counter-Factual datasets, and Feature Representation Confidence (FRC) for each feature.

linguistic features include: *Phonetics/Phonology*: vowels, fricatives, and stress; *Morphology*: noun pluralization and past tense formation (inflectional), and agentive suffixes (derivational); *Syntax*: verb valency (transitive/intransitive), inversion, subject-copula-predicate constructions, and object-complement structures; *Semantics*: simile, metaphor, contrast, sequence, and causality; *Pragmatics*: politeness expressions.

We construct sentences that significantly contain 18 selected linguistic features, called SAELING-DATA. Additionally, we create minimal pairs and counterfactual datasets for these sentences, called SAELING-DATA-MN and SAELING-DATA-CF.

4.2 Main Results

The main experiments to verify that SAELING finds linguistic features in the SAE space and intervening on these features is effective.

4.2.1 Feature Extraction and Analysis

We input the sentences from SAELING-DATA into Llama-3.1-8B and pass the neuron activation distributions after batch normalization through the corresponding SAE layers. We then encode the activation distributions of base vectors at each token in each sentence. The ratio of sentences activated by each base vector is calculated. These base vectors are ranked by this ratio, and their activation is tested on SAELING-DATA-MN and SAELING-DATA-CF. Base vectors that are significantly activated in SAELING-DATA but almost inactive in Datasets SAELING-DATA-MN and SAELING-DATA-CF are selected as potential base vectors for each linguistic feature. The detailed process mode of feature extraction can be found in Appendix B.1.

For each linguistic feature's potential base vectors, we compute their activation in SAELING-DATA, SAELING-DATA-MN, and SAELING-DATA-CF. Using the method described earlier, we calculate the sufficiency and necessity probabilities for each base vector in the minimal pairs and counterfactual datasets, and ultimately compute the FRC score for each feature.

Table 1 shows the activation indices of a representative base vector for each linguistic feature. Due to the varying significance of linguistic features across different sentences, activations corresponding to the base vectors may be lost during the sparse forward propagation process of the SAE, particularly in the TopK activation function mode. We consider a base vector to be strongly related to a linguistic feature if its sufficiency probability exceeds 0.9. For necessity probability, as existing SAEs still exhibit slightly poly-semanticity issues (Olah, 2024), there are still a small amount of low activation for base vectors in positions that do not correspond to the linguistic features. We assume these low activations do not affect the quality of the base vector's representation of the linguistic feature, as we discussed in Appendix B.4.

Overall, the base vectors extracted for features across various linguistic levels show strong correlations. The high quality of phonetic and phonological features indicates that the model contains accurate IPA-related knowledge. The performance of the single feature for metaphor is suboptimal, suggesting that the representation and processing of metaphors may involve more complex mechanisms. Furthermore, FRC values for features in other dimensions mainly exceed 0.9, demonstrating that the selected typical linguistic features consistently yield highly correlated base vectors.

4.2.2 Feature Intervention

We select 5 representative ones for intervention experiments. The intervention method involves

Feature	ID	Enhance		Ablate		FIC
		exp	ctr	exp	ctr	
Morphology						
Past-Tense	8L4016	12.0	4.0	48.0	44.0	8.3
Syntax						
Linking Verb	18L61112	52.0	24.0	48.0	40.0	22.9
Semantics						
Causality	22L53236	32.0	20.0	40.0	36.0	12.0
Simile	26L75327	72.0	52.0	48.0	52.0	6.9
Pragmatics						
Politeness	31L578	60.0	32.0	44.0	20.0	46.9

Table 2: Feature intervention results. The success rates of the extracted linguistic features (Feature, layer, ID) in the enhancement and ablation experiments, along with the final computed FIC score.

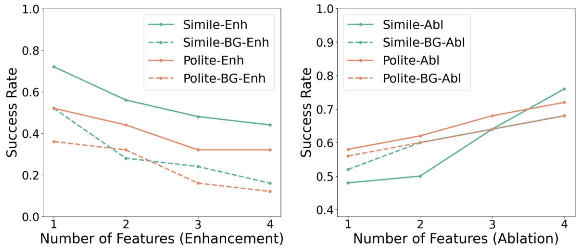

Figure 3: Combined intervention results. Two figures separately present the enhancement and ablation experiment outcomes for the simile and politeness features at layer 26. In these experiments, multiple base vectors corresponding to each feature were jointly intervened.

modifying the activation values of specific base vectors (by index) on a designated SAE layer during forward propagation. We perform two types of intervention: feature enhancement and ablation. With identical input tokens, we set the activation value to 10 for enhancement and 0 for ablation. We then compare the generated outputs with those from the unmodified SAE model, focusing on the prominence of the target linguistic features.

We find that intervening on a single linguistic feature in one layer does not produce effects that are easily distinguishable by human evaluators. Therefore, we use an LLM (GPT-4o) as a judge (Zheng et al., 2023) to assess the prominence of these features in the outputs. For each feature, we conduct 50 experiments and calculate the probabilities of successful enhancement and ablation, *i.e.,* increased and decreased feature prominence, respectively.

In addition, we randomly select 50 base vector indices from the intervention layer and conduct enhancement and ablation experiments under the same conditions as a control. The success rates in the control group are not around 0.5; typically, the enhancement success rate is below 0.5 while the ablation success rate is above 0.5. This discrepancy may stem from the intervention affecting the model's output quality, thereby interfering with the proxy LLM's judgment.

We compute the efficacy of the selected base vectors in both experiments and calculate the FIC value and show the results in Table 2. Our results show that enhancement experiments yield significantly better effects than ablation experiments, with all features demonstrating marked enhancement effects. In the ablation experiments, the politeness feature shows relatively good performance, while other features are less affected; the simile feature does not yield the desired ablation effect. This may be because multiple features in the model control the same linguistic phenomenon. Enhancement interventions have a larger impact on the model, whereas ablation of a single feature may be compensated by other features, leading to suboptimal ablation outcomes. Overall, all 5 features exhibit clear causality in the intervention experiments.

4.3 Analysis

We further conduct analytical experiments to explore the property of SAELING.

4.3.1 Combined Intervention

We find that some layers contain multiple base vectors associated with the same linguistic feature. We can intervene on these base vectors simultaneously to achieve a stronger effect.

We select two linguistic features—simile and politeness—from layer 26. Each feature has four highly related base vectors in this layer. We increase the number of intervened features from one to four. In each experiment, we randomly chose the specified number of base vectors from the four. We used GPT-4o to assess the prominence of the targeted linguistic feature in the generated outputs. For each feature, we conducted 200 enhancement experiments and 200 ablation experiments. We also perform control experiments in layer 26 by randomly selecting a set number of base vectors to intervene.

Figure 3 shows the results for combined intervention. The results indicate that, as the number of intervened base vectors increases, both the directional intervention and the background control

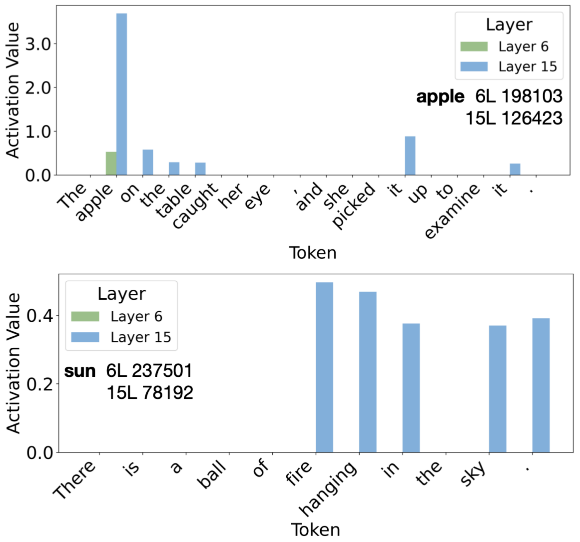

Figure 4: Activation value distributions of deep semantic corresponding features at layer 6 and 15 for reference ambiguity and metaphor example sentences.

experiments exhibit the same trend: the success rate of enhancement experiments decreases, while that of ablation experiments increases. Increasing the number of interventions further affects the quality of the generated text, thereby leading to the observed trend. Moreover, the intervention effect of the feature does not change significantly with an increased number of intervened base vectors, indicating that, after excluding background influences, combined interventions on multiple features in the same layer yield only limited improvement in intervention efficacy.

4.3.2 Deep Semantics Processing

Deep semantics refers to the underlying meaning structures that extend beyond surface-level syntax and lexical definitions. It captures implicit relationships and conceptual associations within language. We conduct experiments to show that SAE

Reference and metaphor exemplify deep semantics by utilizing cognitive mappings and contextual dependencies to convey meaning beyond explicit expression. We conduct experiments on reference and metaphor at the sixth and fifteenth layers respectively. From the results shown on Figure 4, we observe the following:

Reference. In the reference sentence, at the 6$^{\text{th}}$ layer, pronouns do not activate the base vectors corresponding to their referents. At the 15$^{\text{th}}$ layer, pronouns start to activate the correct base vectors (apple) for their referents, effectively resolving reference ambiguity in contexts where multiple possible referents exist. This indicates that as we move deeper into the layers, pronouns generate their deep semantics and disambiguate possible referents.

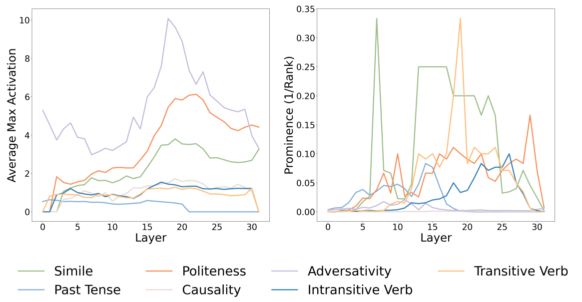

Figure 5: Activation value distributions of deep semantic corresponding features in the 6th and 15th layers for anaphoric and metaphor example sentences.

Metaphor. In the metaphor sentence, our experimental statements contain only the vehicle (fire) and not the tenor (sun). In the 6$^{\text{th}}$ layer, the base vector corresponding to the vehicle is activated, while the base vector for the tenor remains inactive. In the 15$^{\text{th}}$ layer, the activation of the vehicle's base vector decreases, while the base vector for the tenor becomes activated. This suggests that as the model moves to deeper layers, the vehicle maps to the target domain and generates the deep semantics of the tenor, even without the tenor in the context.

4.3.3 Cross-layer Activations

In different layers of the model, we identify distinct base vectors that are activated by the same linguistic feature. To validate this phenomenon, we select a standard base vector from a given layer for each linguistic feature and apply the corresponding SAE to other layers of Llama-3.1-8B. Encoding the hidden states of these layers with the cross-layer SAE, using the same dataset, reveals that in most layers the base vector with the same index as the standard one shows a similar activation pattern on the tokens.

This approach provides an effective tool for observing the cross-layer activation of linguistic features. Figure 5 displays 8 linguistic features. For each feature, the average maximum activation curve and the reciprocal rank curve (used to assess the relative importance of the base vector on the dataset) of the standard base vector are shown for different layers. The distributions indicate that: (1) Linguistic features are widely activated across layers, with only some features failing to activate in the early or late layers; (2) Features that do not activate in the early layers are mostly semantic,

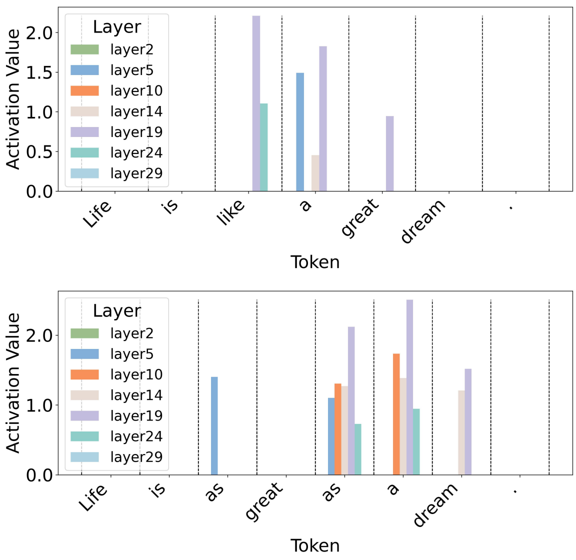

Figure 6: Distribution of average maximum activation values and reciprocal ranks (i.e., the reciprocal of the base vector ranking) for seven typical linguistic features across layers in the dataset.

while those that do not activate in the later layers are mainly syntactic, possibly reflecting the model's functional division across layers; (3) Activation curves typically follow a "rise–plateau–fall" pattern, whereas the reciprocal rank curves often exhibit several sharp peaks. This suggests that each linguistic operation has its own active processing range within the model, although its relative importance may vary across layers.

We further examine the cross-layer activation pattern at the sentence level using the metaphor feature as an example in Figure 6. We select the standard metaphor feature from layer 26 and analyze its activation distribution in layers 2, 5, 10, 14, 19, 24, and 29. In both example sentences, layers 2 and 29 show no token activations, suggesting that the model's initial and final layers may not process metaphors. Layers 5, 10, and 14 appear to be involved in preliminary metaphor processing: in the first sentence, the feature intermittently activated the token following the cue word "like"; in the second sentence, the feature first activated a fixed structure ("as as"), then activated the second "as" and the subsequent token. Around layer 19, the activation extends to the cue word and the entire vehicle, while around layer 24, the activation recedes back to the cue word. These results indicate more complex internal mechanisms for metaphor processing and demonstrate that SAE-based feature extraction is a valuable tool for further exploring the model's internal linguistic mechanisms.

#	Intervene	Model Output
1	**Default**	The wind blows snow into my eyes as I trudge through the blizzard.
	Enhance	As the cold descends, I feel the weight of my breath in my throat. It's an icy haze.
	Ablate	The winter sky was cold. The ice was hard under his boots.
2	**Default**	Love is the burning passion of a summer night.
	Enhance	I feel like butterflies are in my stomach. My heart is beating faster than normal.
	Ablate	The more you write, the more time and love you will have.

Table 3: Case study for intervention under two conditions. Case #1 shows the result when the simile feature is absent from the prompt. Case #2 shows the result when the simile feature is present in the prompt. We highlight spans with simile in the sentences.

4.3.4 Case Study for Intervention

We conduct manual case study on the generated contents after intervening on one identified simile-related base vector. We show cases in Table 3.

In Case #1, the prompt is "*Generate a sentence describing winter*", which does not explicitly include the target linguistic feature. We find that after enhancing the simile-related base vector, the LLM turns to use simile. We can also find that the descriptive and imagistic quality of the default output is stronger than in the ablation results, which indicate that the simile-related base vector is also responsible for vividness.

Case #2 uses the prompt "*Generate a sentence using a simile to describe love*", with explicit requirement for using simile to generate the sentence. When the simile-related base vector is ablated, the LLMs turn to use straightforward descriptions without using similes. Meanwhile, when enhancing the simile-related base vector, the LLMs continue to generate sentences with simile. We show more intervention cases in Appendix C.1.

5 Conclusion

This work addresses two key challenges in interpreting linguistic mechanisms in LLMs: coarse interpretation granularity and insufficient causal analysis. We introduce SAELING, which uses sparse autoencoders (SAE) for fine-grained feature extraction, overcoming poly-semanticity in traditional methods. SAELING also verifies causal relationships by intervening on features, enabling more precise control over model behavior. Our approach reveals that LLMs encode structured linguistic knowledge and offers a robust framework for steering their outputs.

6 Limitations

Our work has several limitations in terms of dataset size, number of features, intervention effects, and cross-layer analysis. Regarding **datasets**, each feature in our study is constructed with approximately 160 sentences. In the future, the dataset can be further expanded to serve as a benchmark for evaluating the language mechanism interpretability of the SAE framework. Concerning the **number of selected features**, we choose 18 representative linguistic features from various theoretical linguistic dimensions. This selection sufficiently demonstrates the effectiveness of our method across different linguistic levels; however, to construct a complete and comprehensive language mechanism system, our approach can be extended to extract a larger number of linguistic features. Achieving this extension will require further work or the development of automated feature extraction methods. In terms of **intervention effects**, our experiments show statistically significant effects for linguistic feature interventions, yet the effect and stability of each case are still inferior to fine-tuning methods. This issue calls for further research and refinement of SAE-based intervention methods. Finally, regarding **cross-layer analysis**, our experiments illustrate the cross-layer mechanism of linguistic features, revealing the potential of our method to explain how large language models process and transmit linguistic information across layers. However, we do not conduct large-scale experiments and inductive analyses in this area, which represents an extension of our method that remains to be explored in future work.

7 Ethical Considerations

This section discusses the ethical considerations and broader impact of this work:

Potential Risks: There is a potential risk that understanding the linguistic mechanisms of the model could provide guidance for embedding malicious information into the model's internal structure. To address this, we will fully open-source our method to enable the community to quickly develop countermeasures in the event of such attacks.

Intellectual Property: The models used, Llama-3.1-8B, and the SAE framework OpenSAE, are both open-source and intended for scientific research use, in accordance with their respective open-source licenses.

Data Privacy: All data used in this research has been manually reviewed to ensure it does not contain any personal or private information.

Intended Use: SAELING is intended to be used as a method for analyzing the mechanisms of large language models.

Documentation of Artifacts: The artifacts, including datasets and model implementations, are comprehensively documented with respect to their domains, languages, and linguistic phenomena to ensure transparency and reproducibility.

AI Assistants in Research or Writing: We employ GitHub Copilot for code development assistance and use GPT-4 for refining and polishing the language in our writing.

References

Zeyuan Allen-Zhu and Yuanzhi Li. 2023. Physics of language models: Part 1, context-free grammar. *ArXiv preprint*, abs/2305.13673.

Aryaman Arora, Dan Jurafsky, and Christopher Potts. 2024. Causalgym: Benchmarking causal interpretability methods on linguistic tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14638–14663, Bangkok, Thailand. Association for Computational Linguistics.

Yuyan Chen, Songzhou Yan, Sijia Liu, Yueze Li, and Yanghua Xiao. 2024. EmotionQueen: A benchmark for evaluating empathy of large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2149–2176, Bangkok, Thailand. Association for Computational Linguistics.

Victoria Fromkin, Robert Rodman, Nina M Hyams, Mengistu Amberber, Felicity Cox, and Rosalind Thornton. 2017. *An Introduction to Language with Online Study Tools 12 Months*. Cengage AU.

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

Jon Gauthier, Jennifer Hu, Ethan Wilcox, Peng Qian, and Roger Levy. 2020. SyntaxGym: An online platform for targeted evaluation of language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 70–76, Online. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, et al. 2024. The llama 3 herd of models. *ArXiv preprint*, abs/2407.21783.

Michael Hahn and Marco Baroni. 2019. Tabula nearly rasa: Probing the linguistic knowledge of character-level neural language models trained on unsegmented text. *Transactions of the Association for Computational Linguistics*, 7:467–484.

Michael Hanna, Sandro Pezzelle, and Yonatan Belinkov. 2024. Have faith in faithfulness: Going beyond circuit overlap when finding model mechanisms. *ArXiv preprint*, abs/2403.17806.

Qianyu He, Sijie Cheng, Zhixu Li, Rui Xie, and Yanghua Xiao. 2022. Can pre-trained language models interpret similes as smart as human? In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7875–7887, Dublin, Ireland. Association for Computational Linguistics.

Jing Huang, Atticus Geiger, Karel D'Oosterlinck, Zhengxuan Wu, and Christopher Potts. 2023. Rigorously assessing natural language explanations of neurons. In *Proceedings of the 6th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 317–331, Singapore. Association for Computational Linguistics.

Robert Huben, Hoagy Cunningham, Logan Riggs, Aidan Ewart, and Lee Sharkey. 2024. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Vivek Iyer, Pinzhen Chen, and Alexandra Birch. 2023. Towards effective disambiguation for machine translation with large language models. In *Proceedings of the Eighth Conference on Machine Translation*, pages 482–495, Singapore. Association for Computational Linguistics.

Shahar Katz and Yonatan Belinkov. 2023. VISIT: Visualizing and interpreting the semantic information flow of transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14094–14113, Singapore. Association for Computational Linguistics.

Boda Lin, Zijun Yao, Jiaxin Shi, Shulin Cao, Binghao Tang, Si Li, Yong Luo, Juanzi Li, and Lei Hou. 2022. Dependency parsing via sequence generation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7339–7353, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Emmy Liu, Chenxuan Cui, Kenneth Zheng, and Graham Neubig. 2022. Testing the ability of language models to interpret figurative language. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4437–4452, Seattle, United States. Association for Computational Linguistics.

Zhu Liu, Cunliang Kong, Ying Liu, and Maosong Sun. 2024. Fantastic semantics and where to find them: Investigating which layers of generative LLMs reflect lexical semantics. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14551–14558. Association for Computational Linguistics.

Kyle Mahowald, Anna A Ivanova, Idan A Blank, Nancy Kanwisher, Joshua B Tenenbaum, and Evelina Fedorenko. 2024. Dissociating language and thought in large language models. *Trends in Cognitive Sciences*.

Christopher D Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, and Omer Levy. 2020. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences*, 117(48):30046–30054.

Aaron Mueller, Garrett Nicolai, Panayiota Petrou-Zeniou, Natalia Talmina, and Tal Linzen. 2020. Cross-linguistic syntactic evaluation of word prediction models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5523–5539, Online. Association for Computational Linguistics.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. 2023. Progress measures for grokking via mechanistic interpretability. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Chris Olah. 2024. The next five hurdles. `https://transformer-circuits.pub/2024/july-update/index.html`.

Harsh Raj, Vipul Gupta, Domenic Rosati, and Subhabrata Majumdar. 2023. Semantic consistency for assuring reliability of large language models. *ArXiv preprint*, abs/2308.09138.

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

Giulia Rambelli, Emmanuele Chersoni, Claudia Collacciani, and Marianna Bolognesi. 2024. Can large language models interpret noun-noun compounds? a linguistically-motivated study on lexicalized and novel compounds. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11823–11835, Bangkok, Thailand. Association for Computational Linguistics.

Subhro Roy, Samuel Thomson, Tongfei Chen, Richard Shin, Adam Pauls, Jason Eisner, and Benjamin Van Durme. 2023. Benchclamp: A benchmark for evaluating language models on syntactic and semantic parsing. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Walid S. Saba. 2023. Stochastic llms do not understand language: Towards symbolic, explainable and ontologically based llms. In João Paulo A. Almeida, José Borbinha, Giancarlo Guizzardi, Sebastian Link, and Jelena Zdravkovic, editors, *Conceptual Modeling*, pages 3–19. Springer Nature Switzerland, Cham.

Hassan Sajjad, Nadir Durrani, and Fahim Dalvi. 2022. Neuron-level interpretation of deep NLP models: A survey. *Transactions of the Association for Computational Linguistics*, 10:1285–1303.

Damien Sileo, Philippe Muller, Tim Van de Cruys, and Camille Pradel. 2022. A pragmatics-centered evaluation framework for natural language understanding. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2382–2394, Marseille, France. European Language Resources Association.

Kevin Stowe, Tuhin Chakrabarty, Nanyun Peng, Smaranda Muresan, and Iryna Gurevych. 2021. Metaphor generation with conceptual mappings. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6724–6736, Online. Association for Computational Linguistics.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024. Language-specific neurons: The key to multilingual capabilities in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Bangkok, Thailand.

THU-KEG. 2025. Opensae: Open-sourced sparse autoencoder towards interpreting large language models.

Yuan Tian, Nan Xu, and Wenji Mao. 2024. A theory guided scaffolding instruction framework for LLM-enabled metaphor reasoning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7738–7755, Mexico City, Mexico. Association for Computational Linguistics.

Lennart Wachowiak and Dagmar Gromann. 2023. Does GPT-3 grasp metaphors? identifying metaphor mappings with generative language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1018–1032, Toronto, Canada. Association for Computational Linguistics.

Leonie Weissweiler, Valentin Hofmann, Anjali Kantharuban, Anna Cai, Ritam Dutt, Amey Hengle, Anubha Kabra, Atharva Kulkarni, Abhishek Vijayakumar, Haofei Yu, Hinrich Schuetze, Kemal Oflazer, and David Mortensen. 2023. Counting the bugs in ChatGPT's wugs: A multilingual investigation into the morphological capabilities of a large language model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6508–6524, Singapore. Association for Computational Linguistics.

Shengguang Wu, Shusheng Yang, Zhenglun Chen, and Qi Su. 2024. Rethinking pragmatics in large language models: Towards open-ended evaluation and preference tuning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22583–22599, Miami, Florida, USA. Association for Computational Linguistics.

Zhengxuan Wu, Thanh-Son Nguyen, and Desmond Ong. 2020. Structured self-AttentionWeights encode semantics in sentiment analysis. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 255–264, Online. Association for Computational Linguistics.

Hanqi Yan, Yanzheng Xiang, Guangyi Chen, Yifei Wang, Lin Gui, and Yulan He. 2024. Encourage or inhibit monosemanticity? revisit monosemanticity from a feature decorrelation perspective. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10423–10435, Miami, Florida, USA. Association for Computational Linguistics.

Akhila Yerukola, Saujas Vaduguru, Daniel Fried, and Maarten Sap. 2024. Is the pope catholic? yes, the pope is catholic. generative evaluation of non-literal intent resolution in llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 265–275.

Kayo Yin and Graham Neubig. 2022. Interpreting language models with contrastive explanations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 184–198, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yan Zhang, Zhaopeng Feng, Zhiyang Teng, Zuozhu Liu, and Haizhou Li. 2023. How well do text embedding models understand syntax? In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9717–9728, Singapore. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

A Dataset Construction

A.1 Linguistic Structure Definition

1. **Phonetics** examines the physical production and acoustic properties of speech sounds.

2. **Phonology** investigates the abstract sound systems and patterns in a language.

3. **Morphology** focuses on the internal structure of words.

4. **Syntax** deals with sentence structure and the rules governing the arrangement of words into phrases and clauses.

5. **Semantics** explores the meaning of words and sentences at a literal or denotational level.

6. **Pragmatics** considers how context influences meaning, covering phenomena like implicature, presupposition, and speech acts.

A.2 Construction Process

When constructing the dataset, we first manually create 5–10 Chinese and English sentences that strongly exhibit the target linguistic feature, along with 5–10 counterfactual sentences. Next, we use GPT-o1 to generate the remaining portion of the dataset, which includes 80 feature-consistent Chinese and English sentences and 80 counterfactual sentences. Minimal pairs are constructed by manually modifying each feature-consistent sentence. After dataset construction, all sentences are manually reviewed and corrected.

A.3 Construction Examples

We present three examples from morphology, syntax, and semantics.

Plural Noun – Minimal Pairs: Directly change the plural form to singular, disregarding grammar.

The books on the shelf are all about history.
The book on the shelf are all about history.
She bought several flowers to decorate the house.
She bought several flower to decorate the house.

Plural Noun – Counterfactual Sentence: Maintain grammatical correctness while removing the plural form.

The books on the shelf are all about history.
The book on the shelf is all about history.
She bought several flowers to decorate the house.
She bought a flower to decorate the house.

Genitive – Minimal Pair: Remove "of" while keeping the rest unchanged.

The roof of the house was damaged in the storm.
The roof the house was damaged in the storm.
The color of the sky changed at sunset.
The color the sky changed at sunset.

Genitive – Counterfactual Sentence: Retain "of" but use a non-genitive context.

The roof of the house was damaged in the storm.
She is afraid of spiders.
The color of the sky changed at sunset.
He ran out of time.

Contrast – Minimal Pair: Remove the contrast marker, disregarding grammar.

I wanted to go out, but it was raining.
I wanted to go out, it was raining.
She was very tired, yet she kept working.
She was very tired, she kept working.

Contrast – Counterfactual Sentence: Change the meaning by altering the logical relation in the second clause to a continuation.

I wanted to go out, but it was raining.
I wanted to go out, then I grabbed an umbrella.
She was very tired, yet she kept working.
She was very tired, then she took a short nap.

A.4 Special Cases for Minimal Pairs

In cases involving transitive verbs, intransitive verbs, linking verbs, and preposed verbs in inversion, direct deletion results in sentences that lose their predicate meaning and cannot convey a complete semantic unit. In such cases, examining activations is meaningless, and minimal contrast pair datasets are not constructed for these features. In the polite speech dataset, the minimal contrast pairs obtained by removing the politeness marker are identical to the counterfactual sentences converted to non-polite sentences; hence, the two datasets are the same.

B Feature Extraction

When extracting features using datasets, we employed the following techniques to improve efficiency and accuracy:

Modes	Simile	Metaphor	Politeness
T-Freq	99	92	37
T-Freq-NN	66	59	26
S-Freq	29	87	12
S-Freq-NN	**13**	**50**	**6**
S-Act	1135	1257	115
S-Act-NN	1121	1242	111

Table 4: Ranking results of different analysis modes: Token Frequency, Token Frequency without noise, Sentence Frequency, Sentence Frequency without noise, Sum Activation, Sum Activation without noise.

B.1 Extraction Modes

We tested the following three analysis modes during feature extraction on the artificially constructed dataset:

Token-Frequency: The total activation frequency of the basis vector across all tokens in the artificial dataset, sorted in descending order.

Sentence-Frequency: The percentage of sentences in the artificial dataset where the basis vector is activated, sorted in descending order.

Sum-Activation: The total activation value of the basis vector across all tokens in the artificial dataset, sorted in descending order.

We build a background dataset with diverse syntactic and semantic patterns. We extract basis vectors that frequently activate across all patterns; their meanings are irrelevant and treated as **background noise**. They can be removed in our experiments.

We test the analysis modes on the three linguistic features—simile, metaphor, and politeness—identified at layer 26. And we extract 42 background noise basis vectors at layer 26.

To evaluate the efficiency of the three analysis modes—*token-frequency*, *sentence-frequency*, and *sum-activation*—we examine the ranking of target basis vectors in the extracted vectors under each analysis mode. We compare the results both with and without the removal of background noise. The results are presented in Table 4. Based on experimental results, performing frequency analysis at the sentence level and removing background noise is most efficient, as it minimizes the impact of irrelevant or outlier activations on the analysis outcomes.

B.2 "20-mix-2" Dataset for Extraction

We first built a dataset composed entirely of sentences highly related to the target linguistic feature. Although directly analyzing sentence-level activation frequencies and sorting the results is feasible, we adopted a more efficient method by inserting 1–2 counterfactual sentences into the dataset. During frequency analysis, we focus on sentences where the counterfactuals show no activation. In such cases, the target feature typically ranks among the top five, greatly enhancing search efficiency.

B.3 Multilingual Dataset

A base vector with robust representational capacity should activate in response to relevant sentences in multiple languages. We observed that many features activate only in one language, which is undesirable. Therefore, we constructed the "20-mix-2" dataset with a 1:1 ratio of English to Chinese sentences, ensuring that the extracted features possess cross-linguistic representational ability.

B.4 Weak Activations

In feature extraction, we observe that some features exhibit weak activation at "interference positions" where the target linguistic feature is not expected. These weak activations occur at positions that either match the cue words of the target feature, share similar morphology with the cue words, or belong to the same linguistic phenomenon category. The weak activations gradually diminish with increasing layer number until they vanish. This indicates that such weak activations are intermediate by-products of the model's internal processing rather than being determined by the inherent connection strength between the base vector and the feature. Therefore, we exclude these weak activations (activations below one-fifth of the maximum activation value) when calculating the necessity probability of a base vector on the dataset. We present examples from morphology, syntax, and semantics as follows:

For the agentive suffix feature (12L248164), consider the following sentences:
(a) The journalist interviewed the mayor about the new policy. The "journalist" token has an activation value of 0.968.
(b) The newspaper published an article about the new policy. The "newspaper" token has an activation value of 0.197.

For the intransitive verb feature (17L63597), consider the sentences:
(a) She traveled to the supermarket. The "traveled" token has an activation value of 1.368.
(b) She drove the car to the supermarket. The

Condition	Politeness	Linking Verb
Enhancement	Can I textbfplease have your email address?	The room should textbfbe large and well lit. It should textbfbe airy and bright and airy.
Default	May I have your phone number?	Sure, my ideal room has good ventilation and textbfis spacious.
Ablation	OK, what is your name?	I can provide you with a list of the ideal characteristics that make up a perfect room.

Condition	Past-Tense
Enhancement	"I was textbfasked for the story. " I having me textbfhad a "one the: " textbftold. They: textbftold:
Default	I'm not a story, I'm a bot.
Ablation	Well, I don't actually have one, and I'm not really sure I'm able to either.

Table 5: Typical outputs from the enhancement, ablation, and default experiments for the politeness, linking verb, and past-tense features.

"drove" token has an activation value of 0.362.

For the simile feature (26L75327), consider the sentences:
(a) Her eyes sparkled like stars in the night sky. The "like" token exhibits an activation value of 3.288.
(b) He looks like his father. The "like" token has an activation value of 0.557.

In these examples, the activations in the (b) sentences are weak and are not considered during feature extraction and evaluation. The presence of weak activation suggests that the model initially activates a broad range of potential semantics and then, during deeper processing, emphasizes the correct, contextually appropriate semantics. This observation warrants further investigation.

C Intervention Experiment Details

C.1 Intervention Cases

We present additional typical cases from other intervention experiments at the Table 5. The prompts used for the three experimental groups are as follows: Politeness: "User: Sir, I want to make an order offline. Assistant:". Linking Verb: "User: Sir, tell me something about your ideal room. Assistant:". Past-Tense: "User: Sir, tell me a story about you. Assistant:".

During manual analysis, both the enhancement and ablation results show clear effects of amplification or suppression of the target linguistic features. Specifically, when intervening with the past tense feature in the 8th layer, the enhancement significantly impacts the coherence of the model's output language. Yet, in the discontinuous output text, the frequency of the morphological past-tense feature still increases dramatically.

C.2 LLM as a Judge

In our feature intervention and combination intervention experiments, we used an LLM as a judge to assess the significance of linguistic features in generated texts. Feature significance is defined based on the frequency, accuracy, and contextual appropriateness of the target feature, as well as its contribution to overall meaning or rhetorical effect.

The prompt structure is as follows:

> **Please compare the following two texts based on {feature}.**
>
> - **Text A**: "{text_a}" - **Text B**: "{text_b}"

Here, `text_a` and `text_b` are generated texts truncated to 100 tokens.

In the intervention experiments, each feature is defined as follows:

Politeness Significance Refers to the degree to which politeness strategies are salient, effective, and contextually integrated. This definition encompasses frequency, pragmatic depth, and social impact in shaping interpersonal rapport, mitigating face threats, and reinforcing cooperative intent.

Past Tense Verb Significance Refers to the degree to which past tense verbs are salient, accurate, and contextually integrated. It includes frequency, morphological consistency, and the rhetorical or narrative impact on establishing a coherent sense of time and providing historical context.

Causality Significance Refers to the degree to which cause-and-effect relationships are clearly indicated, logically structured, and contextually coherent. This includes the frequency and precision of causal connectives (e.g., *because, therefore, thus*) and the depth of reasoning to explain how conditions lead to outcomes.

Linking Verb Structure Significance Refers to the degree to which linking verbs (e.g., *be, become, seem, appear*) are salient, accurate, and contextually integrated. It emphasizes frequency, morphological correctness, semantic clarity, and effectiveness in conveying states, characteristics, or identities.

Simile Significance Refers to the degree to which similes (e.g., comparisons using *like* or *as*) are salient, creative, and contextually integrated. This definition encompasses frequency, imagery richness, and the rhetorical impact on clarity, vividness, and reader engagement.

D Metric Calculation

D.1 Feature Representation Confidence (FRC)

In our feature analysis experiments, we introduce two key causal probabilities that serve as the basis for computing the Feature Representation Confidence (FRC).

The first measure, the *Probability of Necessity* (PN), is defined as $PN = \frac{P(Y=1|do(X=1))-P(Y=1|do(X=0))}{P(Y=1|do(X=1))}$. This metric quantifies the extent to which the presence of a linguistic feature is necessary for the activation of a corresponding base vector. Here, $P(Y = 1 \mid do(X = 1))$ represents the probability that the base vector is activated when the feature is present, whereas $P(Y = 1 \mid do(X = 0))$ indicates the probability of activation when the feature is deliberately suppressed via intervention. The numerator, $P(Y = 1 \mid do(X = 1))-P(Y = 1 \mid do(X = 0))$, captures the net increase in activation due to the feature, and dividing by $P(Y = 1 \mid do(X = 1))$ normalizes this increase relative to the activation when the feature is present.

Similarly, the second measure, the *Probability of Sufficiency* (PS) is expressed as $PS = \frac{P(Y=1|do(X=1))-P(Y=1|do(X=0))}{1-P(Y=1|do(X=0))}$. PS measures the likelihood that the introduction of the feature is sufficient to trigger the activation of the base vector. In this formulation, the denominator $1 - P(Y = 1 \mid do(X = 0))$ represents the maximum possible increase in activation probability (i.e., the probability that the base vector is not activated in the absence of the feature). Thus, PS reflects the proportion of this potential increase that is realized when the feature is present.

Finally, the Feature Representation Confidence (FRC) is computed as the harmonic mean of PN and PS: $FRC = \frac{2\,PN\,PS}{PN+PS}$. The harmonic mean is chosen because it ensures that FRC remains low if either PN or PS is low, thereby providing a balanced measure that only yields a high score when both necessity and sufficiency are strong. This approach allows us to robustly quantify the ability of the SAE latent space's base vectors to represent the targeted linguistic features.

D.2 Feature Intervention Confidence (FIC)

In our methodology, the Feature Intervention Confidence (FIC) score is computed as the harmonic mean of the normalized ablation effect E_{abl} and the normalized enhancement effect E_{enh}:

$$FIC = \frac{2\,E_{abl}\,E_{enh}}{E_{abl} + E_{enh}}.$$

This formulation ensures that FIC is high only when both the ablation and enhancement interventions yield strong effects.

In practice, however, it is possible that one or both of these effects are negative, indicating that an intervention produces an effect opposite to the intended direction. Moreover, even if only one effect is significant while the other is near zero, the feature may still exhibit causal influence. Simply setting an effect that is near zero or negative to 0 would result in an FIC score of 0, which does not adequately capture the underlying causality.

To address this, we introduce a penalty coefficient w to adjust for negative or near-zero effects. Specifically, we define the penalized effect E' for each intervention as follows:

$$E' = \begin{cases} E, & \text{if } E \geq 0, \\ w \cdot |E|, & \text{if } E < 0. \end{cases}$$

Here, w is empirically set to 0.5. Thus, if one of the normalized effects (either E_{abl} or E_{enh}) is negative, we compute its penalized value as 0.5 times its absolute value rather than setting it directly to 0. This approach ensures that even when one of the effects is weak or slightly negative, the FIC score does not vanish entirely, preserving the indication of causality.

Accordingly, the FIC score is then computed as:

$$FIC = \frac{2\,E'_{abl}\,E'_{enh}}{E'_{abl} + E'_{enh}}.$$

In our experiments (see Table2), only the metaphor feature shows a slightly negative ablation effect, while the enhancement and ablation effects for the other features are positive. The introduction of the penalty coefficient w effectively moderates the impact of the negative effect for the metaphor feature, resulting in a more balanced and meaningful FIC score.

This penalty mechanism is crucial because even when only one of the interventions (ablation or enhancement) shows a significant effect, it still provides evidence of the feature's causal role. By incorporating w, we ensure that such cases are not misrepresented by an FIC score of 0, thus offering a more robust measure of the overall causal strength.

E Implementation Details

We used 8 A100 GPUs with 80GB of memory for the experiments. While the exact GPU hours for each experiment were not precisely recorded, the total GPU usage did not exceed one hour. The system was set up with CUDA 12.4, Triton 3.0.0, and Ubuntu 22.04. For the Llama model, we employed the Hugging Face implementation of transformers, and for SAE model, we used the OpenSAE implementation[1] and set the hyperparameter k to 128 for TopK activation.

[1] https://github.com/THU-KEG/OpenSAE