SAFR: Neuron Redistribution for Interpretability

Ruidi Chang **Chunyuan Deng** **Hanjie Chen**
Department of Computer Science
Rice University
{ruidi.chang, hanjie}@rice.edu

Abstract

Superposition refers to encoding representations of multiple features within a single neuron, which is common in deep neural networks. This property allows neurons to combine and represent multiple features, enabling the model to capture intricate information and handle complex tasks. Despite promising performance, the model's interpretability has been diminished. This paper presents a novel approach to enhance model interpretability by regularizing feature superposition. We introduce SAFR,[1] which simply applies regularizations to the loss function to promote monosemantic representations for important tokens while encouraging polysemanticity for correlated token pairs, where important tokens and correlated token pairs are identified via VMASK (Chen and Ji, 2020) and attention weights respectively. We evaluate SAFR with a transformer model on two classification tasks. Experiments demonstrate the effectiveness of SAFR in improving model interpretability without compromising prediction performance. Besides, SAFR provides explanations by visualizing the neuron allocation within the intermediate layers.[2]

1 Introduction

Individual neurons in neural networks can represent multiple features from the input. This phenomenon, known as superposition, improves the model's ability to capture intricate relationships between features (Olah et al., 2020), while also complicating the understanding of the underlying processes behind the model's decision-making (Elhage et al., 2022). Facing these challenges, recent research like sparse autoencoders (SAEs) (Huben et al., 2024) artificially decomposes the activation space into a sparse vector space through auxiliary networks. While SAEs provide a method to *interpret* features through combinations of sparse activations, there is still a lack of sufficient research on *controlling* neuron distribution for interpretability.

In this paper, we ask the question: *Can we enhance model interpretability by explicitly controlling the distribution of features across neurons?* An intuitive approach is to encourage monosemantic neurons by regulating activations (Elhage et al., 2022; Bricken et al., 2023; Wang et al., 2024). However, focusing solely on monosemanticity may limit the model's ability to capture feature interactions, potentially hindering overall performance.

To address this challenge, we propose a novel method called **S**uperposition-**A**ware **F**eature **R**egularization (SAFR) to enhance model interpretability by strategically redistributing neurons through a modified loss function, approached from two perspectives. As illustrated in Figure 1, our framework incorporates regularization techniques aimed at promoting monosemantic representations for important tokens, while simultaneously fostering polysemanticity among correlated token pairs. The identification of important tokens is achieved via a variational inference network, adapted from VMASK (Chen and Ji, 2020). Additionally, correlated token pairs are identified based on attention weights to foster polysemantic representation.

We evaluate the effectiveness of SAFR with a transformer model on the SST-2 (Socher et al., 2013) and IMDB (Maas et al., 2011) datasets. When the top 30% of words identified by SAFR are removed, we observe a significant drop in test accuracy—17.21% on SST-2 and 28.48% on IMDB. As we gradually remove additional words, accuracy continues to decline in a consistent manner. To further substantiate these findings, we visualize the neuron allocation within the FFN layers of a transformer block. Our experimental results demonstrate that SAFR can effectively redistribute features across neurons while preserving a reasonable

[1] SAFR: **S**uperposition-**A**ware **F**eature **R**egularization

[2] The code can be found in https://github.com/chili-lab/SAFR.

Findings of the Association for Computational Linguistics:
NAACL 2025, pages 2117–2126

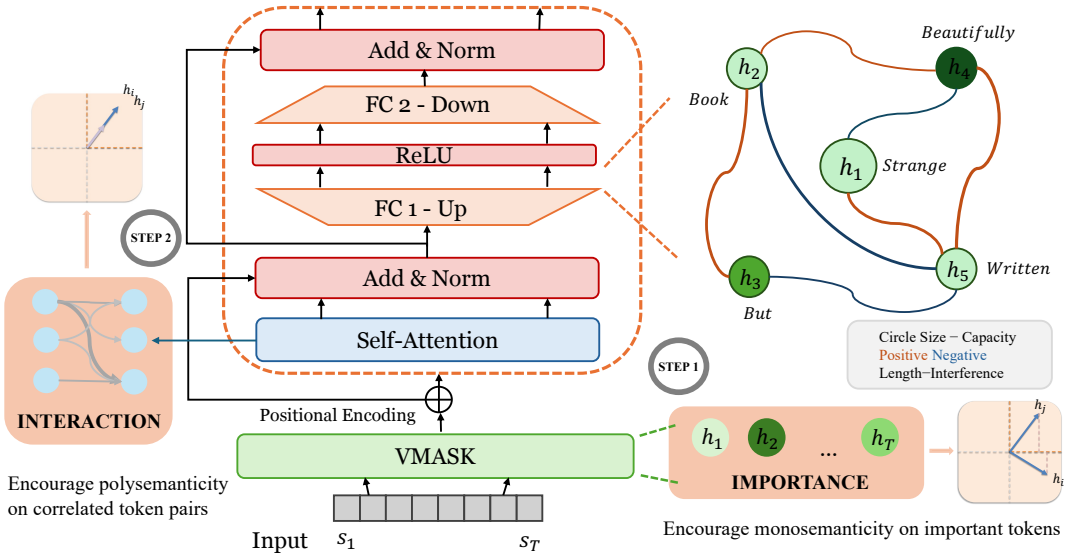


Figure 1: **Basic structure of SAFR**. i) Promote monosemanticity for important tokens after the embedding layers. ii) Leverage the attention mechanism to enhance polysemanticity among correlated token pairs.

degree of polysemanticity, significantly improving interpretability while maintaining prediction performance at comparable levels.

2 Preliminaries

Given an input sequence of T tokens $S = \langle s_1, \ldots, s_T \rangle$ and a neural network model $f(\cdot)$ with L layers, let h_i^ℓ denote the hidden representation obtained at layer $\ell \in [1, \ldots, L]$ and token position $i \in [1, \ldots, T]$. Following previous work (Scherlis et al., 2022; Elhage et al., 2022), which defines *interference*, *polysemanticity*, and *capacity* based on features, we extend these definitions to hidden representations in our analysis.

Interference The *interference* (I) measures the overlap or similarity between two representations at the same layer:

$$I_{i,j}^\ell = h_i^\ell \cdot h_j^\ell \tag{1}$$

Interference quantifies how much the hidden representations interfere with each other. Higher interference indicates greater overlap, suggesting the tokens share representational dimensions.

Polysemanticity The *polysemanticity* (P) describes the extent to which a single hidden representation captures information from multiple tokens:

$$P_i^\ell = \sum_{j \neq i} \left(\hat{h}_i^\ell \cdot h_j^\ell \right)^2 \tag{2}$$

Here, $\hat{h}_i^\ell$ denotes the normalized representation of h_i^ℓ (divided by its own magnitude). A high polysemanticity value indicates that a single direction in the representation space is "polysemantic", meaning it simultaneously represents information from multiple tokens.

Capacity The *capacity* (C) quantifies how much of a hidden representation direction is dedicated to representing token i:

$$C_i^\ell = \frac{(h_i^\ell \cdot h_i^\ell)^2}{\sum_j (h_i^\ell \cdot h_j^\ell)^2} \tag{3}$$

where $0 \leq C_i^\ell \leq 1$ and $1 \leq C^\ell \leq T$ with $C^\ell = \sum_{i=1}^T C_i^\ell$ and T denotes sequence length. A higher capacity value indicates that the representation at position i is more focused on representing the i-th token in the input.

3 Methodology

SAFR enhances model interpretability by strategically redistributing neurons through a superposition regularization strategy. By promoting a well-structured neuron distribution that balances importance and interaction, it makes token representations more meaningful.

The baseline is defined using the original model with cross-entropy loss $\mathcal{L}_{\text{CE}} = -\frac{1}{N} \sum_{n=1}^N \sum_{g=1}^G y_n^g \log p_n^g$, where N is the number of samples in the dataset, and G is the number of classes in text classification, y_n^g is an indicator that equals 1 if sample n belongs to class g (and 0 otherwise), p_n^g is the predicted probability for sample n being of class g. To improve model interpretability in a constrained representational space,

Model	SST-2				IMDB			
	$\text{Acc}_S(\%)$	$\text{Acc}_{\tilde{S}(r)}(\%)$	$\text{Acc}_{\tilde{S}(k)}(\%)$	SRS	$\text{Acc}_S(\%)$	$\text{Acc}_{\tilde{S}(r)}(\%)$	$\text{Acc}_{\tilde{S}(k)}(\%)$	SRS
Baseline	70.21	67.12	66.21	4.00	80.14	77.11	76.60	3.54
$\lambda_{Imp}=0, \lambda_{Inter}=0$	72.56	69.47	64.44	8.12	78.43	76.10	73.56	4.87
SAFR	72.96	70.61	55.75	**17.21**	78.45	75.05	49.97	**28.48**

Table 1: Evaluation on SST-2 and IMDB datasets, k = 30. Acc_S denotes prediction accuracy for original test dataset. $\text{Acc}_{\tilde{S}(r)}$ and $\text{Acc}_{\tilde{S}(k)}$ denote prediction accuracy after randomly deleting $k\%$ of tokens and deleting $k\%$ based on capacity, respectively. SAFR achieves a SRS score of 17.21 for SST-2 and 28.48 for IMDB, outperforming the baseline model, indicating an improvement on model interpretability. The optimal parameter settings for SAFR are $\lambda_{Imp}=0.1, \lambda_{Inter}=0.1$ for SST-2 and $\lambda_{Imp}=0.1, \lambda_{Inter}=1$ for IMDB.

we propose a two-part regularization strategy: one that promotes monosemantic representations for important tokens, and the other encourages polysemanticity for correlated token pairs. This approach enables the model to effectively allocate its representational resources, as shown in Figure 1.

The proposed loss function integrates these regularization terms accordingly:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda_{Imp} \cdot \mathcal{L}_{\text{Importance}} + \lambda_{Inter} \cdot \mathcal{L}_{\text{Interaction}}$$

where λ_{Imp} controls the importance loss term and λ_{Inter} controls the interaction loss term.

Importance-Based Regularization We apply the VMASK (Chen and Ji, 2020) between the embedding layer and the positional encoder to select important tokens. A detailed introduction to VMASK is provided in Appendix A. To encourage monosemanticity for important tokens, we introduce a regularization term $\mathcal{L}_{\text{Importance}} = \frac{1}{T}\sum_{i=1}^{T}\sqrt{P_i^V/E}$, where E represents the embedding dimension and P_i^V denotes the polysemanticity for the hidden representation of the i-th token after the VMASK layer. This regularization penalizes important tokens with high polysemanticity.

Interaction-Based Regularization We leverage the attention mechanism and employ the attention weights to identify correlated tokens. The α-th self-attention head is described as follows:

$$A_\alpha = \text{softmax}\left(\frac{Q_\alpha K_\alpha^T}{\sqrt{d_k}}\right)$$

where $\alpha \in [1, \ldots, M]$, $Q_\alpha = \mathbf{X}W_\alpha^Q$ and $K_\alpha = \mathbf{X}W_\alpha^K$, with $Q_\alpha, K_\alpha \in \mathbb{R}^{T\times d_k}$, d_k denotes the dimension of the key and query vectors. $\mathbf{X} = E_{\text{pos}}(S') \in \mathbb{R}^{T\times E}$ denotes the input matrix to the attention layer, where $E_{\text{pos}}(S')$ is the positional encoding applied to the output S' from the VMASK layer. The score $A_{\alpha(i,j)}$ indicates how much attention token i places on token j.

To encourage highly correlated token pairs to exhibit high polysemanticity, we introduce a loss term $\mathcal{L}_{\text{Interaction}} = \sum_\alpha \sum_{i,j} \frac{1}{T^2} A_{\alpha(i,j)} \cdot (1 - I_{i,j}^{A_\alpha})$, where $I_{i,j}^{A_\alpha}$ is the $Interference$ of the attention weights matrix for the α-th attention head. This loss term penalizes highly correlated tokens that exhibit low interference values.

Proposed Loss Function The loss term is now defined as:

$$\begin{aligned}\mathcal{L} &= \mathcal{L}_{\text{CE}} + \lambda_{Imp} \cdot \mathcal{L}_{\text{Imp}} + \lambda_{Inter} \cdot \mathcal{L}_{\text{Inter}} \\ &= \mathcal{L}_{\text{CE}} + \lambda_{Imp} \cdot \frac{1}{NT} \sum_{n=1}^{N}\sum_{i=1}^{T}\sqrt{\frac{P_i^V}{E}} \\ &\quad + \lambda_{Inter} \cdot \frac{1}{NMT^2} \sum_{n=1}^{N}\sum_{\alpha=1}^{M}\sum_{i,j} A_{\alpha(i,j)}(1 - I_{i,j}^{A_\alpha})\end{aligned}$$

4 Experimental Setup

The proposed method is evaluated on two classification tasks using a standard transformer model.

Datasets We adopt two benchmark datasets: Stanford Sentiment Treebank binary version SST-2 (Socher et al., 2013) and movie reviews IMDB (Maas et al., 2011). Table 4 in Appendix B presents the dataset statistics.

Model We use a typical transformer architecture with a single layer, following the standard setup (Vaswani, 2017). This includes the complete transformer framework with its attention mechanism and positional encoding. In the multi-layer perceptron (MLP) section, we use two fully connected layers: we first expand the dimensionality by a factor of four, apply a ReLU activation, then reduce it back by the same factor, aligning with the commonly used configuration in transformer models. The model uses random embeddings to avoid the influence of pre-trained embedding information. Table 5 in Appendix B presents the model statistics.

Baseline Since our objective is to investigate how regularization can modify neuron resource allocation to enhance interpretability, we employ a standard Transformer model, without any modifications, as the baseline for comparison.

Evaluation To evaluate our method, we define an evaluation metric called **S**uperposition **R**egularization **S**core (**SRS**). By deleting the top $k\%$ of tokens based on capacity, **SRS** calculates the average change in the prediction accuracy over all test data as follows:

$$SRS(k) = \frac{1}{N}\left(\sum_{S=1}^{N} 1\cdot(\hat{y}_S = y) - \sum_{S=1}^{N} 1\cdot(\hat{y}_{\tilde{S}^{(k)}} = y)\right)$$

where $\tilde{S}^{(k)}$ is constructed by dropping the $k\%$ top-scored tokens identified by SAFR from S. The SRS measures how effectively the model arranges neurons. The SRS metric quantifies the alignment between neuron allocations and the semantic significance of tokens. By assessing the structure of neuron allocation, SRS provides insights into the interpretability of the model's internal encoding of information. Higher SRS values indicate that the removed words were highly important, signifying stronger superposition regularization.

5 Results and Analysis

SAFR Improves Interpretability. After applying SAFR, the first fully connected layer redistributes neurons, making *Capacity* interpretable by jointly capturing interference and polysemanticity, thereby reflecting both interaction and importance, as validated by SRS evaluation. Table 1 shows that SAFR achieves SRS scores of 17.21 and 28.48 on the SST-2 and IMDB datasets, outperforming the baseline and demonstrating improved interpretability. Notably, even without regularization ($\lambda_{Imp} = 0$, $\lambda_{Inter} = 0$), the VMASK layer before the Transformer block enhances SRS scores, indicating its positive effect on neuronal allocation. Table 2 presents an ablation study of λ_{Imp} and λ_{Inter}, highlighting two trade-offs: excessive λ_{Imp} may compromise the model's ability to capture feature interactions, while high λ_{Inter} without balancing feature importance can reduce attention to critical tokens. Additional ablation studies are in Appendix D.

	SST-2			
Model	$\text{Acc}_S(\%)$	$\text{Acc}_{\tilde{S}^{(r)}}(\%)$	$\text{Acc}_{\tilde{S}^{(k)}}(\%)$	SRS
Baseline	70.21	67.12	66.21	4.00
$\lambda_{Imp} = 0, \lambda_{Inter} = 0$	72.56	69.47	64.44	8.12
$\lambda_{Imp} = 0.1, \lambda_{Inter} = 0.1$	72.96	70.61	55.75	**17.21**
$\lambda_{Imp} = 0.1, \lambda_{Inter} = 1$	72.67	69.18	59.58	13.09
$\lambda_{Imp} = 1, \lambda_{Inter} = 0.1$	71.81	69.13	56.38	15.43
$\lambda_{Imp} = 100, \lambda_{Inter} = 100$	63.01	61.64	54.20	8.81

Table 2: Ablation study of λ_{Imp} and λ_{Inter}. The choice of λ affects both prediction accuracy and model interpretability.

Sensitivity to k Selection. Figure 3 illustrates the effect of token removal based on capacity. The k% top-scored tokens are directly removed from the original text to ensure that the evaluation reflects the model's ability to perform with reduced input information. As tokens are gradually removed, accuracy declines consistently. The greater decline in our model compared to the baselines suggests better interpretability.

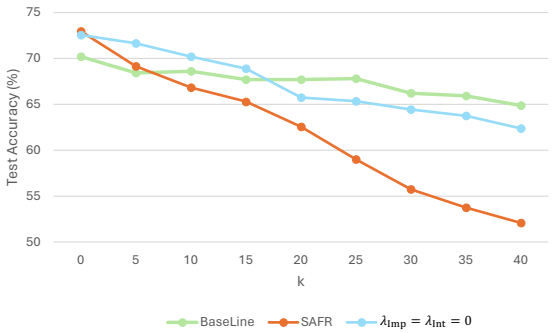
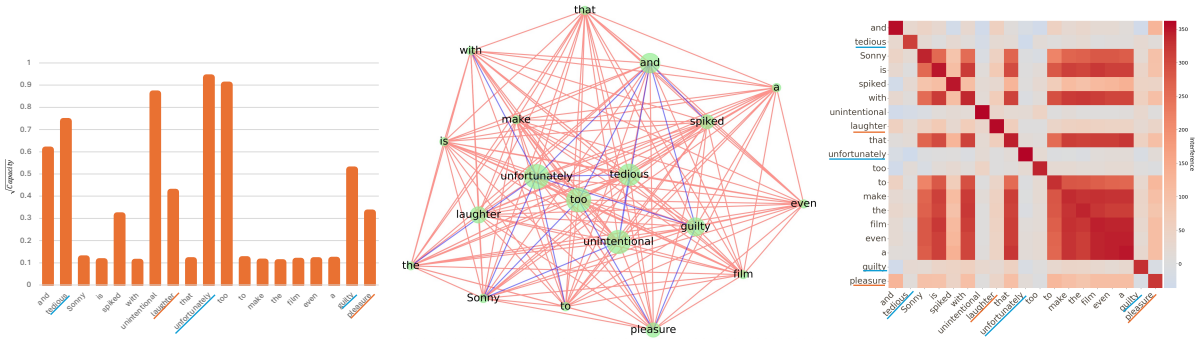


Figure 3: Sensitivity to k Selection. As tokens are gradually removed, accuracy declines consistently.

Average Capacity Across the Vocabulary. Table 3 presents an analysis of the average capacity per token status (important or not w.r.t VMASK) on the entire test set. Tokens are categorized based on VMASK scores, with the top 30% identified as important and the rest 70% as less important. These results demonstrate SAFR's effectiveness in prioritizing and allocating greater representational capacity to task-relevant tokens. This strategic allocation improves the model's clarity in decision-making.

Average Capacity	SST-2	IMDB
All Tokens	0.2981	0.1403
Important Tokens (top 30%)	**0.5745**	**0.2035**
Less Important Tokens (the rest 70%)	0.1794	0.1132

Table 3: Average capacity metric for SST-2 and IMDB datasets. The metric reveals that important tokens exhibit significantly higher capacity scores compared to the overall average capacity.



(a) Capacity Barchart of FC1. (b) Visualization of FC1. (c) Interference Heatmap of FC1.

Figure 2: (a) Important tokens exhibit higher capacity. (b) Circle size represents capacity, with larger circles indicating greater capacity. Red lines denote positive correlations, blue lines indicate negative correlations, and shorter lines indicate stronger correlations. (c) Important tokens demonstrate lower polysemanticity, while correlated token pair exhibit relatively higher interference.

Neuron Allocation Across Layers. We analyzed neuron allocation across layers, with visualizations in Appendix C. Neurons in the embedding layer are randomly distributed, failing to effectively capture their relative importance or inter-neuronal interactions. The VMASK layer begins to identify important tokens at a global level, yet it lacks the capability to analyze token-to-token interactions. While the attention layer demonstrates proficiency in capturing inter-token relationships, the interpretation of token importance remains debated (Jain and Wallace, 2019). The second fully connected layer compresses information into a lower-dimensional space and tend to allocate neuron capacity in a uniform manner. The expansion step in the feedforward network (FC1 layer in our model) allows the network to capture high-dimensional representations of the data. Our observation of this layer reveals that important tokens exhibit greater monosemanticity, and correlated token pairs demonstrate higher interference, illustrated in Figure 2a and 2c.

Visualization of FC1 Figure 2b illustrates the enhanced interpretability achieved after applying SAFR. In the visualization, the size of the circles represents the capacity of individual tokens, where larger circles indicate greater capacity. The red and blue lines depict interference between tokens, with red lines corresponding to positive correlations and blue lines indicating negative correlations. The length of these lines reflects the strength of the correlation. This visualization illustrates neuron distribution and token relationships, demonstrating SAFR's effectiveness in enhancing feedforward layer interpretability. By visualizing these dynamics, the figure highlights key insights into token importance and their interactions within the model.

6 Related Work

Superposition in neural networks has gained attention, with foundational work by (Arora et al., 2018; Goh, 2016). Olah et al. (2020) developed this idea into the "superposition hypothesis" and initiated studies on mechanistic interpretability concerning polysemantic neurons and circuits. Lecomte et al. (2024) showed that polysemanticity can emerge incidentally through regularization and neural noise.

Elhage et al. (2022) illustrated superposition in simplified networks, while subsequent works explored its theoretical, empirical, and applications (Scherlis et al., 2022; Henighan et al., 2023; Hänni et al., 2024; Gurnee et al., 2023; Marshall and Kirchner, 2024; Hänni et al., 2024; Katta, 2024; Chen et al., 2023, 2024).

Interpretability research includes works such as (Dreyer et al., 2024; Black et al., 2022; Wang et al., 2023). Meanwhile, challenges in knowledge (Hu et al., 2024) and identifying universal feature spaces across models (Lan et al., 2024) mark promising directions for future research.

7 Conclusion

In this work, we introduced SAFR, an approach to enhance model interpretability by strategically regularizing feature superposition. Experiments on SST-2 and IMDB show that SAFR improves interpretability, as measured by our SRS metric, without compromising prediction performance.

Our method provides insights into the relationship between superposition and interpretability and offers a framework for visualizing neuron allocation. It contributes to mechanistic interpretability and suggests promising directions for extending the approach to larger models and wider applications.

8 Limitation

This study has several limitations. First, experiments were conducted using a single-layer transformer model; future work should examine the scalability of SAFR with more complex architectures. Second, while focused on classification task, the applicability of SAFR to other NLP tasks—such as natural language inference, question answering, and text generation—remains unexplored. Third, there is a need for more comprehensive and standardized evaluation metrics to assess SAFR effectively. Finally, SAFR does not fully elucidate the causal mechanisms behind the model's decision-making process. Addressing these challenges offers valuable opportunities for future research.

9 Ethic Statements

Our research focuses on understanding and controlling the inner workings of transformer models, without collecting or using any human data; no personal or sensitive information is handled in this study. All datasets used in this work are public.

Acknowledgements

We thank the anonymous reviewers for their valuable comments. We thank the Chili Lab at Rice for helpful discussions and suggestions throughout this research.

References

Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. 2018. Linear algebraic structure of word senses, with applications to polysemy. *Transactions of the Association for Computational Linguistics*, 6:483–495.

Sid Black, Lee D. Sharkey, Léo Grinsztajn, Eric Winsor, Daniel A. Braun, Jacob Merizian, Kip Parker, Carlos Ram'on Guevara, Beren Millidge, Gabriel Alfour, and Connor Leahy. 2022. Interpreting neural networks through the polytope lens. *ArXiv*, abs/2211.12312.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*. Https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Hanjie Chen and Yangfeng Ji. 2020. Learning variational word masks to improve the interpretability of neural text classifiers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4236–4251, Online. Association for Computational Linguistics.

Jiawei Chen, Hongyu Lin, Xianpei Han, Yaojie Lu, Shanshan Jiang, Bin Dong, and Le Sun. 2024. Few-shot named entity recognition via superposition concept discrimination. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7220–7231, Torino, Italia. ELRA and ICCL.

Zhongtian Chen, Edmund Lau, Jake Mendel, Susan Wei, and Daniel Murfet. 2023. Dynamical versus bayesian phase transitions in a toy model of superposition. *ArXiv*, abs/2310.06301.

Maximilian Dreyer, Erblina Purelku, Johanna Vielhaben, Wojciech Samek, and Sebastian Lapuschkin. 2024. Pure: Turning polysemantic neurons into pure features by identifying relevant circuits. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 8212–8217.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. 2022. Toy models of superposition. *arXiv preprint arXiv:2209.10652*.

Gabriel Goh. 2016. Decoding the thought vector.

Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. 2023. Finding neurons in a haystack: Case studies with sparse probing. *Transactions on Machine Learning Research*.

Kaarel Hänni, Jake Mendel, Dmitry Vaintrob, and Lawrence Chan. 2024. Mathematical models of computation in superposition. In *ICML 2024 Workshop on Mechanistic Interpretability*.

Tom Henighan, Shan Carter, Tristan Hume, Nelson Elhage, Robert Lasenby, Stanislav Fort, Nicholas Schiefer, and Christopher Olah. 2023. Superposition, memorization, and double descent. *Transformer Circuits Thread*, 6:24.

Chenhui Hu, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. 2024. Knowledge in superposition: Unveiling the failures of lifelong knowledge editing for large language models. *ArXiv*, abs/2408.07413.

Robert Huben, Hoagy Cunningham, Logan Riggs Smith, Aidan Ewart, and Lee Sharkey. 2024. Sparse autoencoders find highly interpretable features in language

models. In *The Twelfth International Conference on Learning Representations*.

Sarthak Jain and Byron C Wallace. 2019. Attention is not explanation. *arXiv preprint arXiv:1902.10186*.

Pavan Katta. 2024. On implications of scaling laws on feature superposition. *ArXiv*, abs/2407.01459.

Michael Lan, Philip Torr, Austin Meek, Ashkan Khakzar, David Krueger, and Fazl Barez. 2024. Sparse autoencoders reveal universal feature spaces across large language models. *Preprint*, arXiv:2410.06981.

Victor Lecomte, Kushal Thaman, Rylan Schaeffer, Naomi Bashkansky, Trevor Chow, and Sanmi Koyejo. 2024. What causes polysemanticity? an alternative origin story of mixed selectivity from incidental causes. In *ICLR 2024 Workshop on Representational Alignment*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Simon C Marshall and Jan H Kirchner. 2024. Understanding polysemanticity in neural networks through coding theory. *arXiv preprint arXiv:2401.17975*.

Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. 2020. Zoom in: An introduction to circuits. *Distill*. Https://distill.pub/2020/circuits/zoom-in.

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

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Jiachuan Wang, Shimin Di, Lei Chen, and Charles Wang Wai Ng. 2023. Learning from emergence: A study on proactively inhibiting the monosemantic neurons of artificial neural networks. In *Knowledge Discovery and Data Mining*.

Jiachuan Wang, Shimin Di, Lei Chen, and Charles Wang Wai Ng. 2024. Learning from emergence: A study on proactively inhibiting the monosemantic neurons of artificial neural networks. *Preprint*, arXiv:2312.11560.

A VMASK Introduction

VMASK (Chen and Ji, 2020) is a variational word mask layer that is inserted into a neural text classifier and trained with the model. It learns to limit the flow of globally irrelevant or noisy word-level feature information to subsequent network layers, thus forcing the model to focus on the important features for prediction.

B Statistics

This section provides the statistical summaries of the datasets and the model.

Datasets	#Train	#Dev	#Test
SST2	6244	825	1749
IMDB	20k	5k	25k

Table 4: Summary statistics for the datasets, where # counts the number of examples in the train/dev/test sets.

Layer	Dimension
Embedding	(Input Dimension, 256)
FC1	(256, 1024)
FC2	(1024, 256)

Table 5: Summary statistics for the model.

C Neuron Allocation Accross Layers

This section presents the observations regarding neuron allocation across the various layers, as visualized in Figure 4 and 5.

D Regularization Hyperparameter Tuning

This section presents the results of hyperparameter tuning for λ_{Imp} and λ_{Inter}, as summarized in Table 6.

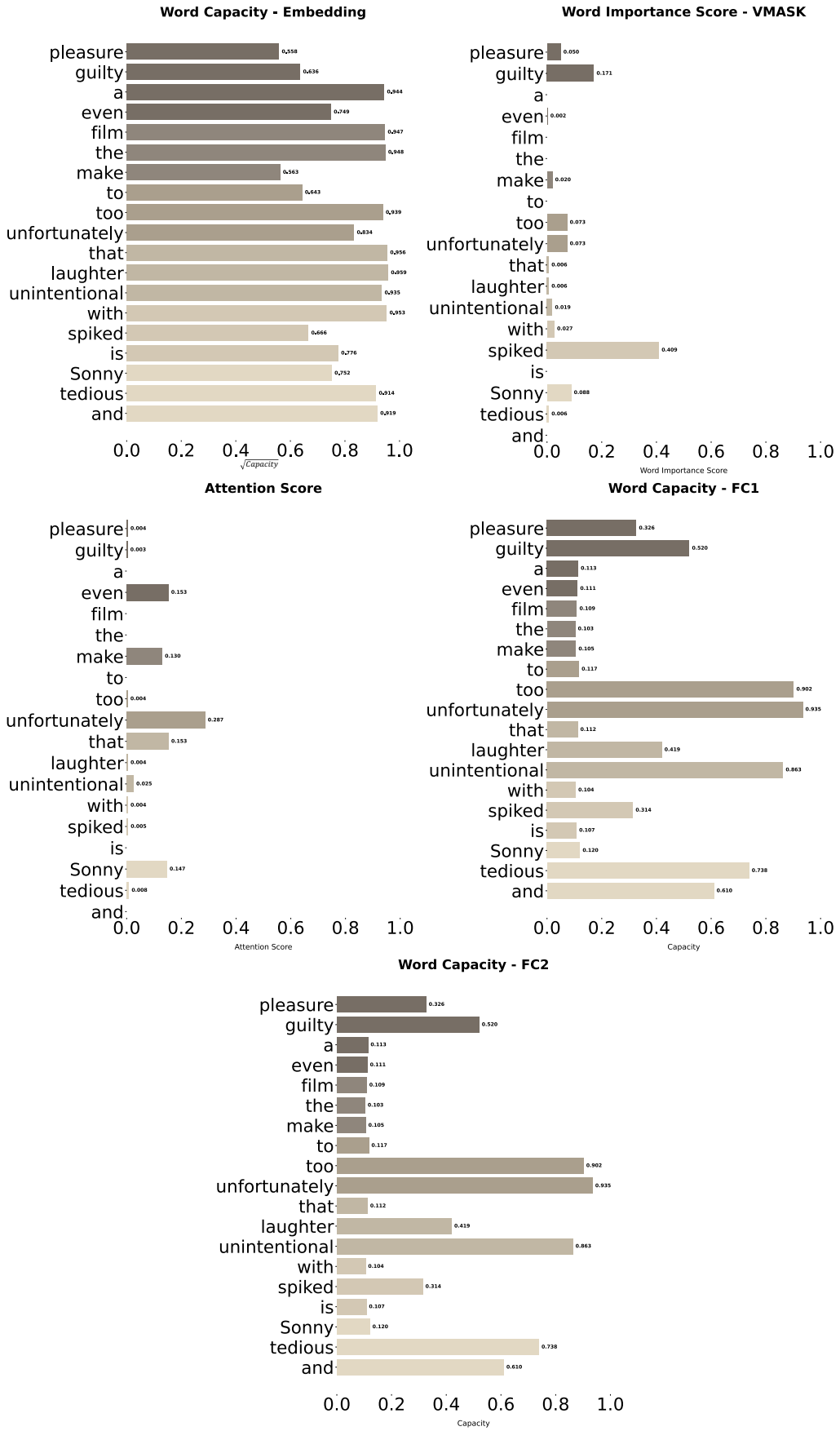


Figure 4: Cross Layers Output: Capacity. VMASK layer uses the importance scores it detects, while the attention layer uses normalized attention scores. The original sentence is “Preposterous and tedious, Sonny is spiked with unintentional laughter that, unfortunately, occurs too infrequently to make the film even a guilty pleasure.”(negative)

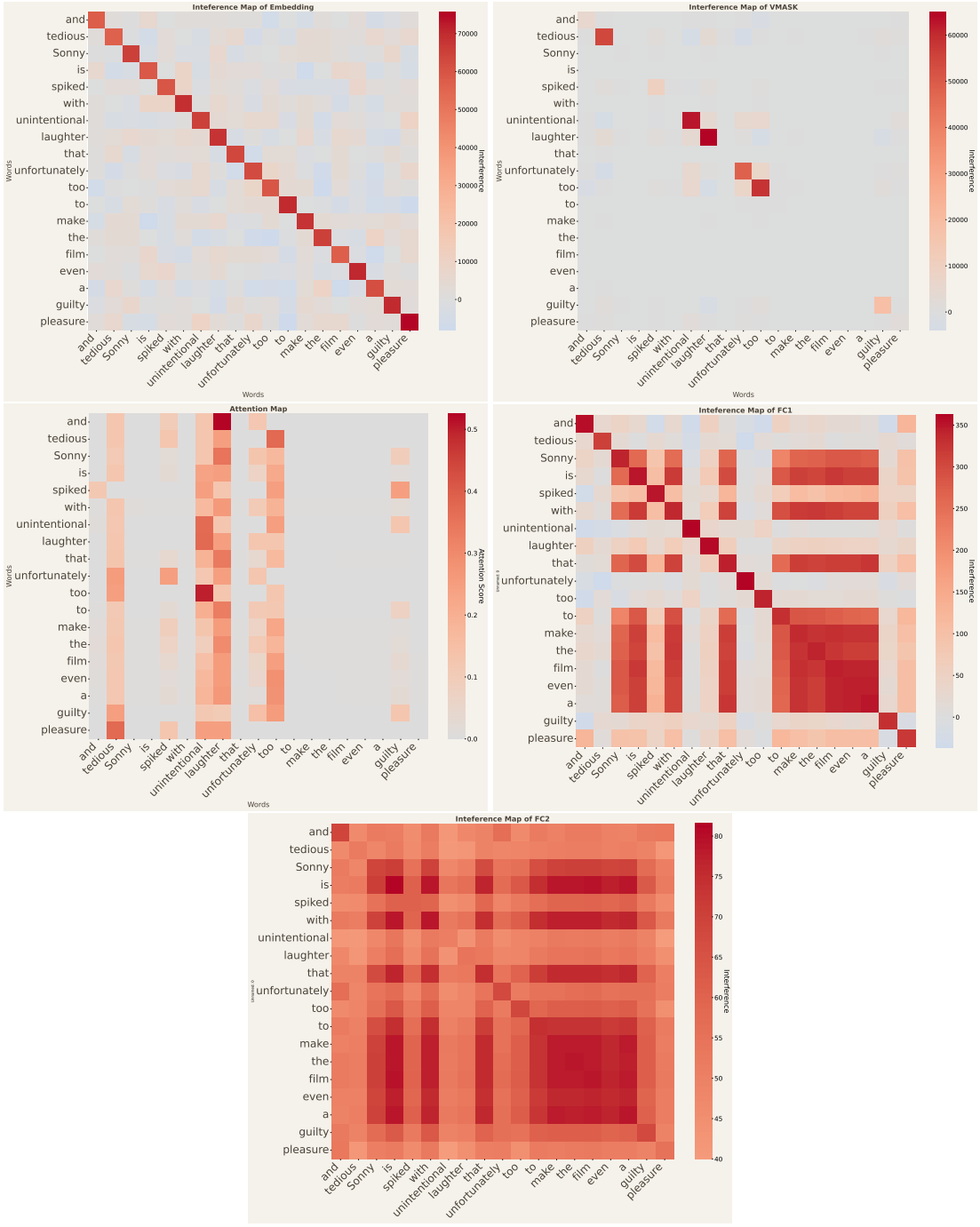


Figure 5: Cross Layers Output: Interference. The attention layer uses the attention weight matrix. The original sentence is “Preposterous and tedious, Sonny is spiked with unintentional laughter that, unfortunately, occurs too infrequently to make the film even a guilty pleasure.”(negative)

Model	SST-2				IMDB			
	$\text{Acc}_S(\%)$	$\text{Acc}_{\tilde{S}^{(r)}}(\%)$	$\text{Acc}_{\tilde{S}^{(k)}}(\%)$	SRS	$\text{Acc}_S(\%)$	$\text{Acc}_{\tilde{S}^{(r)}}(\%)$	$\text{Acc}_{\tilde{S}^{(k)}}(\%)$	SRS
Baseline	70.21	67.12	66.21	4.00	80.14	77.11	76.60	3.54
$\lambda_{Imp} = 0$, $\lambda_{Inter} = 0$	72.56	69.47	64.44	8.12	78.43	76.10	73.56	4.87
$\lambda_{Imp} = 0$, $\lambda_{Inter} = 0.01$	72.84	70.27	62.78	10.06	81.16	78.20	74.61	6.55
$\lambda_{Imp} = 0$, $\lambda_{Inter} = 1$	73.01	69.07	60.32	12.69	80.44	76.89	71.84	8.60
$\lambda_{Imp} = 0$, $\lambda_{Inter} = 100$	59.35	57.75	56.38	2.97	77.83	74.90	67.64	10.19
$\lambda_{Imp} = 0.01$, $\lambda_{Inter} = 0$	72.67	69.87	61.41	11.26	80.00	77.02	70.98	9.02
$\lambda_{Imp} = 0.01$, $\lambda_{Inter} = 0.01$	73.07	71.07	58.49	14.58	80.30	77.42	64.58	15.72
$\lambda_{Imp} = 0.1$, $\lambda_{Inter} = 0.1$	72.96	70.61	55.75	17.21	79.24	75.96	53.99	25.25
$\lambda_{Imp} = 0.1$, $\lambda_{Inter} = 1$	72.67	69.18	59.58	13.09	78.45	75.05	49.97	28.48
$\lambda_{Imp} = 1$, $\lambda_{Inter} = 0$	71.30	66.96	55.06	16.24	74.95	72.30	50.07	24.88
$\lambda_{Imp} = 1$, $\lambda_{Inter} = 0.1$	71.81	69.13	56.38	15.43	74.89	71.72	50.08	24.81
$\lambda_{Imp} = 1$, $\lambda_{Inter} = 1$	71.93	70.67	56.78	15.15	74.78	71.44	53.94	20.84
$\lambda_{Imp} = 100$, $\lambda_{Inter} = 0$	64.67	63.75	57.92	6.75	51.53	51.20	50.00	1.53
$\lambda_{Imp} = 100$, $\lambda_{Inter} = 100$	63.01	61.64	54.20	8.81	53.48	53.33	50.00	3.48

Table 6: Ablation Study on SST-2 and IMDB datasets. The choice of λ affects both prediction accuracy and model interpretability.