Exploring Activation Patterns of Parameters in Language Models

Yudong Wang, Damai Dai, Zhe Yang, Jingyuan Ma, Zhifang Sui

School of Computer Science
State Key Laboratory of Multimedia Information Processing
Peking University
{yudongwang, mjy}@stu.pku.edu.cn, {daidamai, yz_young, szf}@pku.edu.cn

Abstract

Most work treats large language models as black boxes without an in-depth understanding of their internal working mechanism. To explain the internal representations of LLMs, we utilize a gradient-based metric to assess the activation level of model parameters. Based on this metric, we obtain three preliminary findings. (1) When the inputs are in the same domain, parameters in the shallow layers will be activated densely, which means a larger portion of parameters will have great impacts on the outputs. In contrast, parameters in the deep layers are activated sparsely. (2) When the inputs are across different domains, parameters in shallow layers exhibit higher similarity in the activation behavior than in deep layers. (3) In deep layers, the similarity of the distributions of activated parameters is positively correlated to the empirical data relevance. Further, we develop three validation experiments to solidify these findings. (1) Firstly, starting from the first finding, we attempt to configure different sparsities for different layers and find this method can benefit model pruning. (2) Secondly, we find that a pruned model based on one calibration set can better handle tasks related to the calibration task than those not related, which validates the second finding. (3) Thirdly, Based on the STS-B and SICK benchmarks, we find that two sentences with consistent semantics tend to share similar parameter activation patterns in deep layers, which aligns with our third finding. Our work sheds light on the behavior of parameter activation in LLMs, and we hope these findings will have the potential to inspire more practical applications.

Code — https://github.com/Qian2333/Exploring-Activation-Patterns-of-Parameters-in-Language-Models

Introduction

Since the emergence of GPT-4 (Achiam et al. 2023), there has been a surge of interest in Large Language Models (LLMs). As these LLMs continue to advance and their capabilities strengthen, there remains a noticeable gap in research dedicated to their interpretability.

The study aims to investigate the coexistence of different capabilities within the model. More specifically, when faced with inputs across various domains, we observe variations in the internal representation of LLMs. There have been some explorations into the functions of specific layers and parameters in LLMs (Azaria and Mitchell 2023; Geiger et al. 2024). It is generally recognized that some significantly different capabilities of LLMs cannot fully coexist within a limited scale. However, no targeted research analyzes the operational patterns of specific capabilities within language models in a more general sense.

Recent work has found that some parameters within the model exist for specific tasks. Fu et al. (2023) revealed that while the distilled model excels in the specific task it was designed for, the original, more general model experiences a decline in performance in other tasks previously proficient in. This observation suggests that different tasks may tap into distinct capacities within a model, and these capacities seem to be mutually exclusive to some extent. In another study, Zhang et al. (2021) introduced the notion that LLMs inherently evolve into a Mixture of Experts (MoE) within themselves. This concept implies that different parts of the network are tasked with handling different inputs, further strengthening the idea of internal specialization within the model.

Building on the insights from the aforementioned phenomena, this study seeks to unravel the following questions: Which parameters within the network are activated to determine the outputs, and does the distribution of these activated weights exhibit distinct patterns when faced with inputs across different domains? In essence, we aim to explore whether the degree of parameter activation in different layers varies in response to non-homogenous input scenarios and if so, to what extent.

Drawing on methods from network pruning (Ma, Fang, and Wang 2023), we assess the influence of a parameter by comparing the original output of a model to that of a model in which the parameter is set to 0. Specifically, we employ the first-order term of the Taylor expansion of the model to gauge the impact of a parameter on the outputs. Given two sentences s_1, s_2, whether homogenous or not, we derive two vectors V_{s_1}, V_{s_2} that characterize the influence of the internal parameters of the model. By examining the **cos**ine similarity between these two vectors from **LLM** with different **d**ata (LLMDcos), we observe three phenomena:

- For inputs in the same domain, parameters in the shallow layers of the model are activated densely, while parameters in the deep layers are activated sparsely.

- For inputs from different domains, the similarity of the activation patterns of parameters in the shallow layers of the model is higher than in deep layers.
- In deep layers, the similarity of the distributions of activated parameters is positively correlated to the empirical data relevance.

To validate our observed results, we design three experiments, which include model pruning and semantic similarity tasks. The improved pruning method based on our analytical results outperforms the original. We validated our second finding by comparing the performance changes caused by the different calibration sets of the pruning method. The proposed LLMDcos is also validated to be related to semantic similarity.

Our contributions are listed in the following:

- We observe the different capabilities of different layers in LLMs, summarize three phenomena, and design experiments to validate each.
- We optimize the pruning method, providing a reference for other pruning methods.
- We propose a new method for calculating data similarity based on gradient information.

Background and Motivation

Motivation

Our motivation for this study stems from a phenomenon observed in distillation (Fu et al. 2023): when a specific capability of a general model is enhanced through distillation, there tends to be a corresponding decline in other evaluations. This observation prompts us to investigate the nature of the relationship between different capabilities within a model - are they mutually reinforcing or mutually exclusive? And if both, under what circumstances does each scenario occur?

A study (Zhang et al. 2021) proposed the idea that a model internally generates a Mixture of Experts, which suggests that the model handling of different tasks could be attributed to the spontaneous formation of a sparse structure during training. This structure, in turn, might harbor distinct capabilities that are mutually exclusive to some extent. Same idea has also been mentioned in several works (Huang et al. 2021; Xia et al. 2023; Wang et al. 2023; Razdaibiedina et al. 2023).

Our work is primarily related to model pruning. The parameter scoring method from model pruning serves as a valuable tool to explore which parameters within the model are most responsive to a given input. Meanwhile, the variation in the model's performance during evaluation due to different pruning settings can also validate our conclusions.

Model Pruning

The crux of model pruning lies in identifying the crucial parameters within the network. From the perspective of model pruning, we can derive insights into the significant role scoring of parameters.

Model pruning techniques (LeCun, Denker, and Solla 1989; Hassibi, Stork, and Wolff 1993; Han et al. 2015) for LLMs can be broadly categorized into two types (Zhu et al. 2023): structured pruning (Frantar and Alistarh 2023; Zhang et al. 2023; Sun et al. 2023) and unstructured pruning (Santacroce et al. 2023; Ma, Fang, and Wang 2023). Structured pruning aims to reduce the hidden state size by removing entire rows or columns from the weight matrix, which can lead to actual acceleration and pruning benefits. Unstructured pruning, on the other hand, involves eliminating individual connections, i.e., specific elements within the weight matrix. This approach can maintain model performance even at high pruning ratios but does not inherently lead to computational speedup unless a substantial proportion of connections is pruned within specific regions.

Regardless of the type of pruning, both methods focus on identifying which components of the network have the least impact on the output. Many studies (LeCun, Denker, and Solla 1989; Ma, Fang, and Wang 2023; Frantar and Alistarh 2023) have utilized Taylor expansion to define the rank of weights in terms of their influence on the network's structure, thereby guiding the pruning process by removing weights with minimal impact.

Preliminary Findings

In all the content of this paper, unless specifically marked, all model results are analysis results of Llama2-7B (Touvron et al. 2023). This paper provides results from more models in subsequent sections.

Definition of Activation and LLMDcos

We begin with a standard unstructured model pruning problem in an empirical scenario. Given a trained model, unstructured model pruning attempts to rank the parameters within the model and drop them (set them to 0). Prior research has shown that retraining the model after pruning (Han et al. 2015) can lead to better performance. However, for LLM, the performance of the pruned model remains competitive even without retraining (Ma, Fang, and Wang 2023; Sun et al. 2023). Considering the substantial resources required for retraining, we only prune the model without retraining in this work.

For ranking the parameters, the activation level of the ith parameter w_i within the model is defined in various ways. Han et al. (2015) defines the activation by magnitude, removes the parameters with magnitude below a threshold, and then retrains the model. Paul et al. (2022) train a weight as the activation level for each parameter. In this work, we consider a gradient-based activation (Frantar and Alistarh 2023; Ma, Fang, and Wang 2023), but simplify it to only the first order. Here, given a sentence s, the activation of the ith parameter in the model w_i is defined as

$$\begin{aligned}\mathcal{A}(s, w_i) &= |D(s, w_i) - D(s, 0)| \\ &= |w_i \cdot \frac{\partial D(s, w_i)}{\partial w_i} + O(w_i^2)| \\ &\approx |w_i \cdot \frac{\partial D(s, w_i)}{\partial w_i}|\end{aligned}$$

where $D(s, w_i)$ refers to the logit for the next token with max probability.

By concatenating all the $\mathcal{A}(w_i)$, we derive $\mathcal{A}(w) \in R^n$, where n denotes the number of parameters within the model.

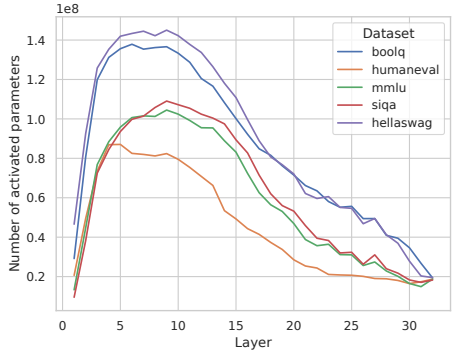


Figure 1: Activated parameter statistics for domain-specific tasks. The x-axis represents the layer, and the y-axis represents the number of the parameter w_i satisfying $\mathcal{A}(w_i) > 0.00002$.

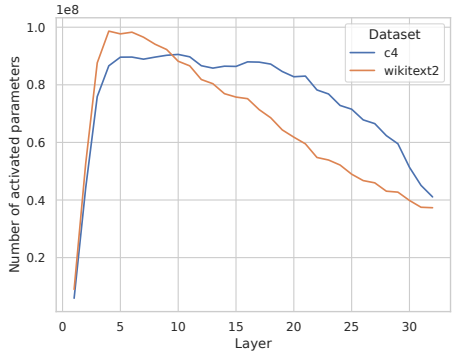


Figure 2: Activated parameter statistics for general corpora, including C4 and wikitext2. The x-axis represents the layer, and the y-axis represents the number of the parameter w_i satisfying $\mathcal{A}(w_i) > 0.00002$.

We examine $\mathcal{A}(w)$ across different sentences from various data sources. We define the **cos**ine similarity metric of a **D**ata pair based on the **L**arge **L**anguage **M**odel (LLMDcos):

$$LLMDcos(s_1, s_2) = \frac{\mathcal{A}(s_1, w) \cdot \mathcal{A}(s_2, w)}{\sqrt{\|\mathcal{A}(s_1, w)\|^2 \cdot \|\mathcal{A}(s_2, w)\|^2}}$$

Where s_1, s_2 are two sentences. With LLMDcos, we aim to analyze different layers within LLMs, as well as the differences between various inputs.

Finding 1: Parameter Activation Patterns for Inputs in the Same Domain

In this section, we attempt to analyze the distinct behaviors within Large Language Models (LLMs). Using our defined $\mathcal{A}(w_i)$, we analyze the distribution of activated parameters in different layers when faced with a single input.

In Figure 1 and Figure 2, we present the statistical results from two data sources: 1) specific tasks including Boolq (Clark et al. 2019a), MMLU (Hendrycks et al. 2020), HumanEval (Chen et al. 2021), SIQA (Sap et al. 2019), and hellaswag (Zellers et al. 2019), 2) general corpora, including C4 (Raffel et al. 2020) and Wikitext2 (Merity et al. 2016). For each dataset, we select 64 samples and averaged the activation status of each parameter facing different samples. When calculating the activation status of data in different layers, we collectively consider the parameters within a MLP layer. Specifically, this includes parameters from seven parts: the fully connected linear layers of Q, K, V, O, and the three fully connected linear layers of the MLP layer.

For domain-specific tasks, from the statistical results in Figure 1, we can find that fewer parameters are activated in the first layer, meaning that only a small portion of parameters have a significant impact on the results. In the shallow layers, the parameters that have a greater influence ($\mathcal{A}(w_i) > 0.00002$) on the results gradually increase. In the relatively deeper layers, the parameters that have a significant impact on the results decrease, concentrating on specific parts. This phenomenon is consistent across the five data sets representing different abilities. This leads us to speculate that for a single task, apart from the first layer, many parameters in the shallow layers are involved in the proceeding of the results. Conversely, in the deep layers and the first layer, fewer parameters have a significant impact on the results.

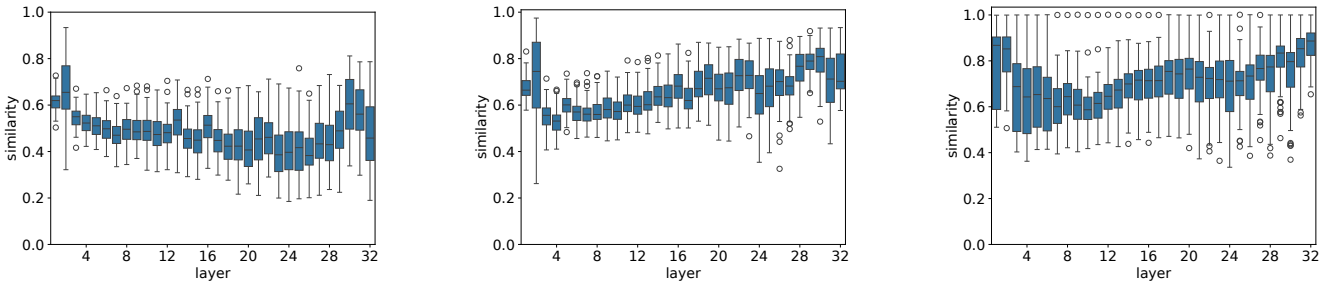
For general corpora, the results remain similar to those of domain-specific tasks in the shallow layers but differ significantly in the deep layers. When dealing with general corpora, the average activation of the parameters in the deep layers is higher than that in specific tasks. This is particularly evident in C4, which shows less specialization than Wikitext2.

Finding 2: Parameter Activation Patterns for Inputs in the Different Domains

To observe the functionality of different layers within Large Language Models (LLMs), we have documented the activation scenarios of various layers when faced with input from different data domains. In response to inputs from two data domains, we calculated the LLMDcos for each layer each time.

As depicted in Figure 3, we analyze data from three data domains: Boolq, HumanEval, and MMLU. Apart from HumanEval-HumanEval where we only experimented with 16 samples, we statistically analyzed 64 samples in all other experiments. Consistent with the previous section, within the same layer, we included the parameters from seven parts in our statistics.

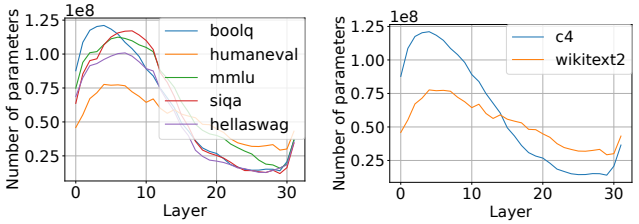
From the results, we observed that facing the same data source, the activation levels of different layers were similar. When faced with diverse data sources, the first 12 layers of the network had higher LLMDcos, while the later layers had lower similarity. Notably, the second layer had higher similarity in any analysis between the two data sources. Moreover, when faced with empirically similar data sources, the similarity in activation levels of the later layers was relatively high, that is, MMLU and Boolq showed relatively high similarity, while the activation distribution of both datasets had a significant difference from the HumanEval dataset. Combining the conclusions from the previous section, we can speculate that for different tasks, the shallow end of the network has



(a) Boolq-Boolq (b) MMLU-MMLU (c) HumanEval-HumanEval

(d) Boolq-MMLU (e) Boolq-HumanEval (f) MMLU-HumanEval

Figure 3: Statistics of LLMDcos for different layers in Llama2-7b ($whis = 1.5$). The x-axis represents the layer number, and the y-axis represents the LLMDcos values of 64 samples. The two data sources are indicated below the image.



(a) domain-specific tasks (b) general corpora

Figure 4: Activated parameter statistics on Llama3-8B. The x-axis represents the layer, and the y-axis represents the number of the parameter w_i satisfying $\mathcal{A}(w_i) > 0.00002$.

a more general understanding ability. When faced with various tasks, similar parameters are activated to understand the problem, especially in the second layer. In the deeper layers, the network has relatively dispersed parameters, that is, some parameters are activated for specific tasks.

Finding 3: Observing Data through Activated Distribution

Through the above experiments, we observed that for different datasets, the similarity of the deep layers in the model significantly decreases, while for the same dataset, the similarity between the shallow and deep layers of the model remains consistent. This leads us to hypothesize that the similarity in the later layers may be related to semantic similarity. To validate this hypothesis, we test the similarity of more datasets in the Section Validation 3 and conduct tests on benchmarks for semantic similarity.

Generality Across Different Models

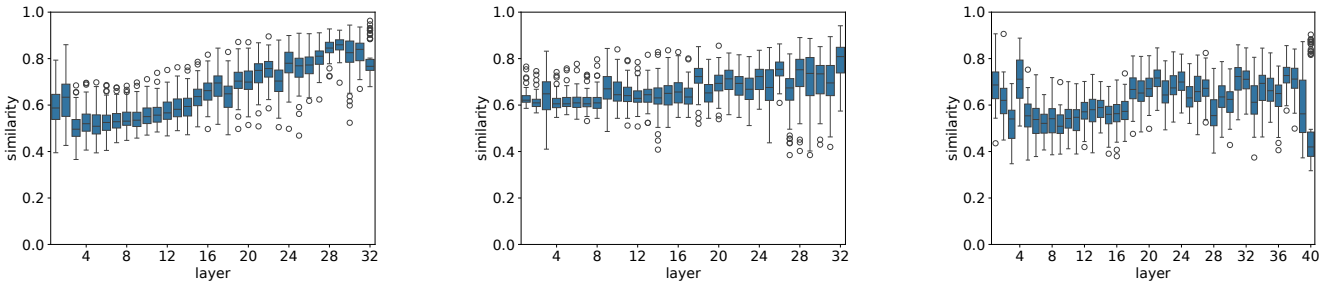
To determine the universality of our results, rather than their specificity to the Llama series or the 7B models, we performed experiments analogous to those in the following section on Qwen-7B (Bai et al. 2023), Llama2-13B (Touvron et al. 2023) and Llama3-8B (AI@Meta 2024). The results are illustrated in the Figure 4 and Figure 5. We can find the patterns among all the models containing the same tendency with higher activation levels and LLMDcos in the shallow layers and reverse in the deep layers.

Validation Experiments

This paper, drawing upon the preceding analysis, suggests several application approaches. The success of these applications substantiates the validity of our analytical results. None of the pruning experiments include retraining.

Validation 1: Pruning LLMs with Different Sparasity based on Activation Level

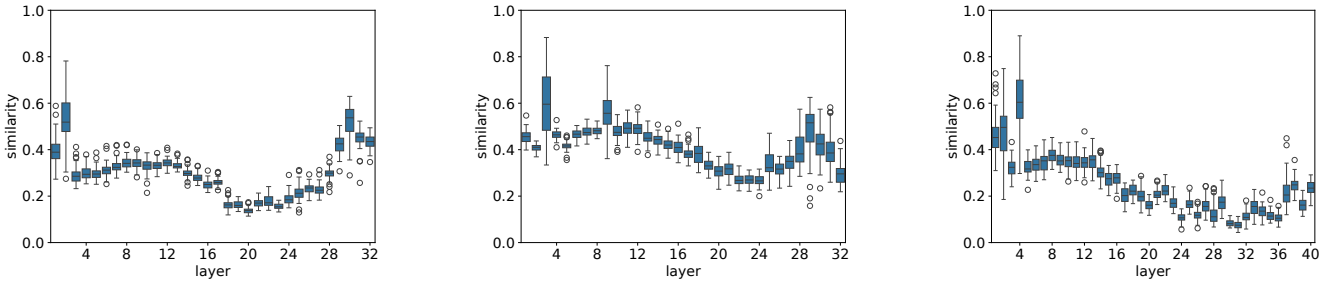
Based on the analysis results from Finding 1, we can observe the following phenomena: For a 32-layer Llama2-7B network, most of the parameters in the 1-2 layers and the deep half of the network have little impact on the results. However, in the parameters of the 3-17 layers of the network, there are relatively more parameters that have a significant impact on the results. To validate this conclusion, we will prune the model, making the 3-17 layers of the network less sparse,



(a) Llama3-8B MMLU-MMLU

(b) Qwen-7B MMLU-MMLU

(c) Llama2-13B MMLU-MMLU



(d) Llama3-8B MMLU-HumanEval

(e) Qwen-7B MMLU-HumanEval

(f) Llama2-13B MMLU-HumanEval

Figure 5: Statistics of LLMDcos values for different models ($whis = 1.5$). The x-axis represents the layer number, and the y-axis represents the LLMDcos values of 64 samples. The two data sources and the model are indicated below the image. The images are consistent with the results from Llama2-7B.

while the 1-2 and 18-32 layers of the network have a higher degree of sparsity.

We employed the unstructured pruning method proposed by (Sun et al. 2023) without retraining, maintaining all other settings constant on Llama-7B, Llama2-7B, Llama2-13B, Llama3-8B (AI@Meta 2024). For a language model with L layers, given the total sparsity S and the number of activated parameters ($\mathcal{A}(w_i) > \gamma$) N_i for each ith layer, we set the sparsity of the ith layer S_i as

$$N_i' = sigmoid(\frac{N_i - mean(N_i)}{std(N_i)} + \frac{\alpha}{i})$$

$$S_i = S + \beta S(1 - \frac{N_i' L}{\Sigma_{i=1}^{L} N_i'})$$

We set $\alpha = 2, \beta = 0.2$ for all the experiments. We set $\gamma = 0.00002$ for all models with 32 layers, while $\gamma = 0.000005$ for 40-layers (Llama-2-13B).

Here, we set an heuristic term $\frac{\alpha}{i}$ to balance the great gap between the 1-2 layers and deeper layers which make higher sparsity in deeper layers. We use the sigmoid function to maintain a certain amount of parameters during pruning. While the pruning method (Wanda) cannot precisely determine parameters for specific utility, it is essential to maintain a moderate level of sparsity.

To compare the test results of the pruned networks, we conducted tests on two different metrics on six datasets based on the original settings. All the calibration dataset is C4 (Raffel et al. 2020). The evaluation tasks include Wikitext2 (Merity et al. 2016), Boolq, SIQA (Sap et al. 2019), PIQA (Bisk et al. 2020), Hellaswag (Zellers et al. 2019), MMLU.

As shown in Table 1, our method consistently improves the perplexity on Wikitext2 across all model results, suggesting that our approach can generally enhance the language modeling capability of the models. In the zero-shot results, it is worth noting that the improvements in Hellaswag are universal, while Boolq generally experiences a decline. In conjunction with Figure 6, we find that Hellaswag has the highest correlation with C4, while Boolq is relatively lower. Therefore, the results of using C4 as the calibration set are less satisfactory in Boolq.

Validation 2: Pruning LLMs with Different Calibration Set

Based on Finding 2, we observe that: For Llama-2-7B, faced with different data domains, most parameters in the deep part of the network exhibit a lower degree of activation similarity. In contrast, in the shallow layers of the network, there is a relatively higher degree of similarity in the distribution of parameter activation. This leads us to hypothesize that the shallow layers of the network consist of more generic parameters, while the deep layers are discretely composed of parameters that address different problems. To validate this conclusion, we will modify the calibration set to specialized tasks, with the pruning method in Validation 1.

Models	pruning method	wikitext2↓	zero-shot				MMLU
			Boolq	SIQA	PIQA	Hellaswag	
Llama-7B	Dense	5.68	71.48	38.28	52.73	28.13	32.03
	Wanda	7.26	66.41	35.94	**52.73**	29.30	20.70
	Wanda(ours)	**7.19**	**70.70**	35.94	48.05	**30.86**	**28.52**
Llama2-7B	Dense	5.12	76.95	54.30	53.13	34.77	42.19
	Wanda	6.46	**77.73**	40.23	51.56	29.30	**37.11**
	Wanda(ours)	**6.39**	76.95	**43.36**	**52.34**	**35.55**	33.98
Llama2-13B	Dense	4.57	80.86	62.50	55.86	53.91	48.05
	Wanda	5.58	**81.64**	55.47	53.91	50.00	42.58
	Wanda(ours)	**5.55**	80.86	**57.03**	**59.77**	**50.78**	**43.75**
Llama3-8B	Dense	5.54	64.45	68.36	63.28	43.36	51.95
	Wanda	9.06	**63.67**	52.34	52.34	27.34	28.13
	Wanda(ours)	**8.69**	63.28	**50.39**	**53.13**	**31.25**	**30.08**

Table 1: Pruning experiment results. All models have an overall sparsity of 50%, with c4 as the calibration set. The evaluation metric for wikitext2 is perplexity, and for MMLU it is 5-shot. Wanda (ours) refers to our Wanda model after layer-by-layer adjustment of the pruning ratio, ensuring the overall pruning ratio remains the same.

Models	Calibration set	wikitext2↓	zero-shot				MMLU
			Boolq	SIQA	PIQA	Hellaswag	
Llama2-7B	MMLU	**6.49**	73.83	**42.19**	52.34	**33.20**	**35.16**
	SIQA	6.69	**74.10**	41.02	**52.73**	30.86	34.77
Llama2-13B	MMLU	**5.63**	**80.08**	54.30	55.08	50.00	**44.92**
	SIQA	5.77	77.73	**55.08**	**55.08**	**51.17**	40.23
Llama3-8B	MMLU	**9.25**	63.28	56.64	53.52	34.77	**34.77**
	SIQA	9.84	**63.28**	**56.64**	**53.52**	**35.16**	29.30

Table 2: Pruning experiment results. All models have an overall sparsity of 50%, and the pruning method used is Wanda (ours). The evaluation metric for wikitext2 is perplexity, and for MMLU it is 5-shot.

We adopted the pruning method from Validation 1: Wanda (ours), ensuring all other settings remained unchanged. We adjusted the calibration set to MMLU and SIQA. To ensure the sentence length of the calibration set is the same, we concatenated different MMLU and SIQA data into long sentences and cut them to a specific length (2048 for the Llama-2-7B, 4096 for the Llama-2-13B, and 8096 for Llama-3-8B).

The results are shown in the Table 2. We observe that compared to the C4 dataset, the language modeling capability (as measured by PPL in Wikitext2) of the pruning results using SIQA as the calibration set is consistently lower. This might be due to the greater diversity of MMLU. The pruning results using MMLU as the calibration set consistently outperform those based on C4 and SIQA on MMLU. Moreover, on SIQA, PIQA, and Hellaswag, three evaluations with stronger correlation with SIQA, the pruned model results using SIQA as the calibration set always outperform those of MMLU. It excepts Llama2-7B, according to the lower modeling capability. This further validates our hypothesis.

Validation 3: Semantic Similarity with LLMDcos

In the analysis presented in Finding 3, we observed that the similarity of the deep layers of the network decreases for different data domains.

To validate the relationship between LLMDcos and data relevance, we tested its performance on a semantic similarity benchmark STS-B (Cer et al. 2017) and SICK (Marelli et al. 2014). Each sample of STS-B and SICK consists of two sentences and a simlilar score as label.

As shown in Table 3, the LLMDcos calculated by Llama2-7B yielded the best results. As evident from the results, LLama2-7b achieved the best performance, while the results of Llama2-13B were relatively inferior. We believe that, compared to semantic similarity, LLMDcos assess more of the similarity in the capabilities of LLMs required by the inputs. For 13B models, these capabilities may be more densely represented in the parameters, leading to these deviations.

In addition to this, we calculated the similarity relationships across nine datasets, including Boolq, C4, GSM8K (Cobbe et al. 2021), Hellaswag, HumanEval, MMLU, PIQA, SIQA, and Wikitext2. The results are shown in Figure 6.

Discussion

Worse Performance of Pruned Llama-3-8B

Refer to Table 1 and Table 2, although the dense Llama-3-8B shows better performance on SIQA, PIQA, and MMLU, it fails to maintain this superior performance after pruning. Con-

Model	STS-B	SICK
Llama-7B	0.30	0.52
Llama2-7B	0.66	0.51
Llama2-13B	0.43	0.52

Table 3: Spearman correlation between LLMDcos and semantic similarity. We sampled 256 examples on each dataset ($p \leq 0.001$). We use the LLMDcos of 20-30 layers (20-38 for Llama2-13B)

sidering the extensive training tokens (15T) (AI@Meta 2024) for Llama-3-8B, it is evident that Llama-3-8B is trained in a manner that better specializes its capabilities within specific internal parameters as shown in Figure 4a. This phenomenon aligns with our findings, which indicate that pruning leads to a loss of capability outside the calibration domain. As shown in Table 2, Llama-3-8B pruned with SIQA remains competitive on SIQA evaluations among all models.

Extreme Performance of the First Layers and Last Layers in Findings

In Figure 1 and Figure 3, the first two layers and the last two layers perform in an extreme pattern.

For the extremely low numbers observed in Finding 1, we notice that both parts are close to the readable tokens. This could be caused by common words and the specific language (English) since all the LLMs evaluated in this paper are multilingual.

As for the slight increase observed in the last layers in Finding 2, this could be due to the specific answers in the datasets, as both Boolq and MMLU consist of selected questions.

Related Work

Causal Inference

In line with our intention, many researches in causal inference aim to explore the mechanisms and patterns within the network. These studies employ probes (Hupkes, Veldhoen, and Zuidema 2018; Peters et al. 2018; Tenney, Das, and Pavlick 2019; Clark et al. 2019b), attribution methods (Shrikumar et al. 2016; Sundararajan, Taly, and Yan 2017), causal alignment (Geiger et al. 2021, 2024), and some even more drastic measures (Zhao, Li, and Sun 2023), such as directly skipping specific layers to observe the difference in results.

However, most of the work focuses on a specific function rather than a macroscopic discussion of the functions of different layers. Zhao et al. (Zhao, Li, and Sun 2023) found that the third layer in Llama may significantly impact whether the model outputs toxic information. Azaria et al. (Azaria and Mitchell 2023) verified that the later layers of the model are more aware of whether they contain false information. Mcgrath et al. (McGrath et al. 2023) indicated that different layers in the model may have various information about numbers.

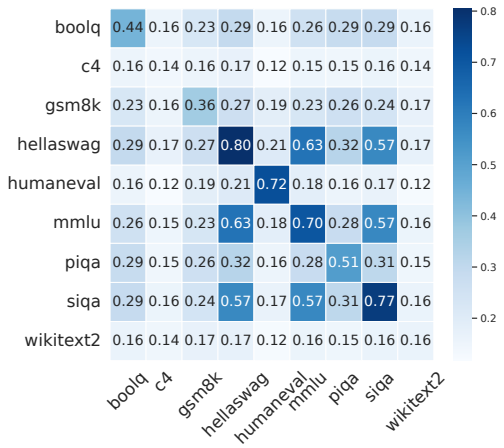


Figure 6: Dataset relevance. The figure shows the similarity calculated based on the mean LLMDcos of layers 20-30 in Llama2-7B.

Data Similarity

Our work reflects data relevance. In the field of NLP, data similarity mainly refers to text similarity. Text similarity in early machine learning was primarily based on statistical methods, such as word frequency and sentence length. However, in the era of deep learning, semantic similarity has become of greater interest, leading to the proposal of a series of models.

Most of the related recent work has directly calculated relevance using the hidden state after the embedding layer. (Laskar, Huang, and Hoque 2020) proposed two methods for calculating data similarity, one based on the average pooling of the token dimension of BERT's hidden state and the other through training with the CLS token. Both methods assisted in the main task of answer selection. (Li et al. 2020) improved the effect of similarity calculation by mapping data similarity to a Gaussian distribution based on a kernel distribution.

Limitations

We identify two main limitations of this study: GPU limitations and the lack of theoretical proof.

The GPU memory limitations prevent us from verifying the pruning results of Llama2-70B.

In terms of theory, our activation degree algorithm is a very rough calculation. More refined calculations are constrained by time and space complexities and cannot be performed on our machine.

Conclusion

This paper explores the activation patterns of internal parameters in language models and subsequently introduces LLMDcos to calculate the similarity of activation patterns when facing inputs from different domains. Based on this exploration, we make three findings reflecting the inner patterns that vary among different layers. Various experiments are conducted to validate the findings.

Acknowledgements

This paper is supported by NSFC project 62476009. The contact author is Zhifang Sui.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

AI@Meta. 2024. Llama 3 Model Card.

Azaria, A.; and Mitchell, T. 2023. The internal state of an llm knows when its lying. *arXiv preprint arXiv:2304.13734*.

Bai, J.; Bai, S.; Chu, Y.; Cui, Z.; Dang, K.; Deng, X.; Fan, Y.; Ge, W.; Han, Y.; Huang, F.; et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Bisk, Y.; Zellers, R.; Bras, R. L.; Gao, J.; and Choi, Y. 2020. PIQA: Reasoning about Physical Commonsense in Natural Language. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Cer, D.; Diab, M.; Agirre, E.; Lopez-Gazpio, I.; and Specia, L. 2017. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*.

Chen, M.; Tworek, J.; Jun, H.; Yuan, Q.; de Oliveira Pinto, H. P.; Kaplan, J.; Edwards, H.; Burda, Y.; Joseph, N.; Brockman, G.; Ray, A.; Puri, R.; Krueger, G.; Petrov, M.; Khlaaf, H.; Sastry, G.; Mishkin, P.; Chan, B.; Gray, S.; Ryder, N.; Pavlov, M.; Power, A.; Kaiser, L.; Bavarian, M.; Winter, C.; Tillet, P.; Such, F. P.; Cummings, D.; Plappert, M.; Chantzis, F.; Barnes, E.; Herbert-Voss, A.; Guss, W. H.; Nichol, A.; Paino, A.; Tezak, N.; Tang, J.; Babuschkin, I.; Balaji, S.; Jain, S.; Saunders, W.; Hesse, C.; Carr, A. N.; Leike, J.; Achiam, J.; Misra, V.; Morikawa, E.; Radford, A.; Knight, M.; Brundage, M.; Murati, M.; Mayer, K.; Welinder, P.; McGrew, B.; Amodei, D.; McCandlish, S.; Sutskever, I.; and Zaremba, W. 2021. Evaluating Large Language Models Trained on Code.

Clark, C.; Lee, K.; Chang, M.-W.; Kwiatkowski, T.; Collins, M.; and Toutanova, K. 2019a. BoolQ: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Clark, K.; Khandelwal, U.; Levy, O.; and Manning, C. D. 2019b. What does bert look at? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*.

Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; Hesse, C.; and Schulman, J. 2021. Training Verifiers to Solve Math Word Problems. *arXiv preprint arXiv:2110.14168*.

Frantar, E.; and Alistarh, D. 2023. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *International Conference on Machine Learning*, 10323–10337. PMLR.

Fu, Y.; Peng, H.; Ou, L.; Sabharwal, A.; and Khot, T. 2023. Specializing smaller language models towards multi-step reasoning. In *International Conference on Machine Learning*, 10421–10430. PMLR.

Geiger, A.; Lu, H.; Icard, T.; and Potts, C. 2021. Causal abstractions of neural networks. *Advances in Neural Information Processing Systems*, 34: 9574–9586.

Geiger, A.; Wu, Z.; Potts, C.; Icard, T.; and Goodman, N. 2024. Finding alignments between interpretable causal variables and distributed neural representations. In *Causal Learning and Reasoning*, 160–187. PMLR.

Han, S.; Pool, J.; Tran, J.; and Dally, W. 2015. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28.

Hassibi, B.; Stork, D. G.; and Wolff, G. J. 1993. Optimal brain surgeon and general network pruning. In *IEEE international conference on neural networks*, 293–299. IEEE.

Hendrycks, D.; Burns, C.; Basart, S.; Zou, A.; Mazeika, M.; Song, D.; and Steinhardt, J. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Huang, Y.; Zhang, Y.; Chen, J.; Wang, X.; and Yang, D. 2021. Continual learning for text classification with information disentanglement based regularization. *arXiv preprint arXiv:2104.05489*.

Hupkes, D.; Veldhoen, S.; and Zuidema, W. 2018. Visualisation and'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. *Journal of Artificial Intelligence Research*, 61: 907–926.

Laskar, M. T. R.; Huang, X.; and Hoque, E. 2020. Contextualized embeddings based transformer encoder for sentence similarity modeling in answer selection task. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 5505–5514.

LeCun, Y.; Denker, J.; and Solla, S. 1989. Optimal brain damage. *Advances in neural information processing systems*, 2.

Li, B.; Zhou, H.; He, J.; Wang, M.; Yang, Y.; and Li, L. 2020. On the sentence embeddings from pre-trained language models. *arXiv preprint arXiv:2011.05864*.

Ma, X.; Fang, G.; and Wang, X. 2023. Llm-pruner: On the structural pruning of large language models. *Advances in neural information processing systems*, 36: 21702–21720.

Marelli, M.; Menini, S.; Baroni, M.; Bentivogli, L.; Bernardi, R.; and Zamparelli, R. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, 216–223. Reykjavik, Iceland: European Language Resources Association (ELRA).

McGrath, T.; Rahtz, M.; Kramar, J.; Mikulik, V.; and Legg, S. 2023. The hydra effect: Emergent self-repair in language model computations. *arXiv preprint arXiv:2307.15771*.

Merity, S.; Xiong, C.; Bradbury, J.; and Socher, R. 2016. Pointer Sentinel Mixture Models. arXiv:1609.07843.

Paul, M.; Chen, F.; Larsen, B. W.; Frankle, J.; Ganguli, S.; and Dziugaite, G. K. 2022. Unmasking the Lottery Ticket Hypothesis: What's Encoded in a Winning Ticket's Mask? *arXiv preprint arXiv:2210.03044*.

Peters, M. E.; Neumann, M.; Zettlemoyer, L.; and Yih, W.-t. 2018. Dissecting contextual word embeddings: Architecture and representation. *arXiv preprint arXiv:1808.08949*.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140): 1–67.

Razdaibiedina, A.; Mao, Y.; Hou, R.; Khabsa, M.; Lewis, M.; and Almahairi, A. 2023. Progressive prompts: Continual learning for language models. *arXiv preprint arXiv:2301.12314*.

Santacroce, M.; Wen, Z.; Shen, Y.; and Li, Y. 2023. What matters in the structured pruning of generative language models? *arXiv preprint arXiv:2302.03773*.

Sap, M.; Rashkin, H.; Chen, D.; LeBras, R.; and Choi, Y. 2019. Socialiqa: Commonsense reasoning about social interactions. *arXiv preprint arXiv:1904.09728*.

Shrikumar, A.; Greenside, P.; Shcherbina, A.; and Kundaje, A. 2016. Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*.

Sun, M.; Liu, Z.; Bair, A.; and Kolter, J. Z. 2023. A simple and effective pruning approach for large language models. *arXiv preprint arXiv:2306.11695*.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. In *International conference on machine learning*, 3319–3328. PMLR.

Tenney, I.; Das, D.; and Pavlick, E. 2019. BERT rediscovers the classical NLP pipeline. *arXiv preprint arXiv:1905.05950*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Wang, L.; Li, L.; Dai, D.; Chen, D.; Zhou, H.; Meng, F.; Zhou, J.; and Sun, X. 2023. Label words are anchors: An information flow perspective for understanding in-context learning. *arXiv preprint arXiv:2305.14160*.

Xia, M.; Gao, T.; Zeng, Z.; and Chen, D. 2023. Sheared llama: Accelerating language model pre-training via structured pruning. *arXiv preprint arXiv:2310.06694*.

Zellers, R.; Holtzman, A.; Bisk, Y.; Farhadi, A.; and Choi, Y. 2019. HellaSwag: Can a Machine Really Finish Your Sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Zhang, M.; Shen, C.; Yang, Z.; Ou, L.; Yu, X.; Zhuang, B.; et al. 2023. Pruning meets low-rank parameter-efficient fine-tuning. *arXiv preprint arXiv:2305.18403*.

Zhang, Z.; Lin, Y.; Liu, Z.; Li, P.; Sun, M.; and Zhou, J. 2021. Moefication: Transformer feed-forward layers are mixtures of experts. *arXiv preprint arXiv:2110.01786*.

Zhao, W.; Li, Z.; and Sun, J. 2023. Causality analysis for evaluating the security of large language models. *arXiv preprint arXiv:2312.07876*.

Zhu, X.; Li, J.; Liu, Y.; Ma, C.; and Wang, W. 2023. A survey on model compression for large language models. *arXiv preprint arXiv:2308.07633*.