A Closer Look into Mixture-of-Experts in Large Language Models

Anonymous ACL submission

Abstract

Mixture-of-experts (MoE) is gaining increasing attention due to its unique properties and remarkable performance, especially for language tasks. By sparsely activating a subset of parameters for each token, MoE architecture could increase the model size without sacrificing computational efficiency, achieving a better trade-off between performance and training costs. However, the underlying mechanism of MoE still lacks further exploration, and its modularization degree remains questionable. In this paper, we make an initial attempt to understand the inner workings of MoE-based large language models. Concretely, we comprehensively study the parametric and behavioral features of three recent MoE-based models and reveal some intriguing observations, including (1) Neurons act like fine-grained experts. (2) The router of MoE usually selects experts with larger output norms. (3) The expert diversity increases as the layer increases, while the last layer is an outlier. Based on the observations, we also provide suggestions for a broad spectrum of MoE practitioners, such as router design and expert allocation. We hope this work could shed light on future research on the MoE framework and other modular architectures.

1 Introduction

The advent of Large Language Models (LLMs) revolutionized the field of Natural Language Processing. LLMs researchers are continually pushing the boundaries of Language Models by scaling up both model size and the column of training data, significantly enhancing the capabilities of these models. This escalation in training cost and complexity necessitates innovative solutions to better balance between pre-training efficiency and model performance. One emerging solution to this end is the Mixture-of-Experts (MoE) (Shazeer et al., 2017) architecture. The MoE framework facilitates the computational efficiency of the model by dynamically routing inputs to a subset of experts, allowing

for substantial model scaling while maintaining training costs and leading to numerous influential advancements in the field (Reid et al., 2024; Jiang et al., 2024; Dai et al., 2024; Team, 2024).

Beyond efficiency, another attractive trait of MoE architecture is its modular design and learning paradigm. This modularization could enable more flexible and potentially more generalizable handling of diverse data and tasks within a single MoE model by assigning them to more specialized experts. Despite its widespread adoption, it remains an open question whether existing MoE-based LLMs truly leverage this modularity in their knowledge distribution or model behaviors across different experts. In other words, is MoE a simple ensemble of homogeneous experts or a modular combination of heterogeneous experts? Answering the above question comprehensively is non-trivial. Therefore, in this paper, we take the first step by investigating three recent MoE-based LLMs (Mixtral 8x7B (Jiang et al., 2024), DeepSeekMoE (Dai et al., 2024), and Grok-1[1]) from two critical perspectives: model parameters and model behaviors. We aim to explore common and distinct features and behaviors among different experts, further shedding light on the inner mechanisms of MoE-based models.

Specifically, we examine the correlation between experts' parameters, gates, and their output features given text inputs. Before diving into deeper analyses, we briefly summarize some of our empirical conclusions (detailed in Sec 6) and observations:

- Neurons in the Feed-Forward Networks (FFN) layer are fine-grained experts. Both the gate embedding and the gate projection matrix for experts perform the choosing operation: the former determines the expert selection while the latter controls the neuron activation. Meanwhile, we observe that their similar heat maps exhibit correlation, revealing that from the gate

---

[1] https://github.com/xai-org/grok-1

projection perspective, the expert neurons can probably be regarded as 'tiny' experts with one neuron.

- Increasing the number of experts in deeper layers but reducing it in the last layer. We observe that the similarities between experts' parameters and outputs continuously decrease with the increase in layer number while suddenly increasing in the last layer.
- Employing the norm as the routing mechanism is a reasonable choice. For Mixtral and DeepSeek models, we find that the gate usually selects experts with larger output norms.
- When analyzing the correlation between experts, measuring the similarities between weight matrices is, to some extent, equivalent to measuring the averaged similarities of expert outputs.
- Compared with some special initialization schemes, training MoE from scratch is more likely to facilitate expert diversity. This stems from the observations that there exist stronger correlations (*e.g.*, higher similarities) between Mixtral experts' parameters and behaviors, while DeepSeek and Grok, which are trained from scratch, do not show such correlations.

2 Preliminary: Mixture-of-Experts

Mixture-of-Experts models enhance transformers by replacing the original feed-forward networks (FFNs) with N parallel FFNs combined with a router. These N FFNs are also called experts and denoted as E_n for $n \in [1, N]$. The router $g(\cdot; \mathbf{G}, \mathrm{k})$, parameterized by $\mathbf{G}$ and an integer k, assigns the input $\mathbf{x}$ to a score distribution over the experts, $g(\mathbf{x}; \mathbf{G}, \mathrm{k}) \in \mathbb{R}^N$. Typically, the router g consists of a simple linear layer followed by a softmax and a Top-k function.

Given $\mathbf{x} \in \mathbb{R}^h$, the output $\mathbf{y} \in \mathbb{R}^h$ is the weighted sum of the outputs from all experts:

$$\mathbf{y} = \sum_{n \in N} g_n(\mathbf{x}; \mathbf{G}, \mathrm{k}) E_n(\mathbf{x})$$

When k for Top-k is smaller than N, only a subset of experts is involved in the computation. This is known as Sparse Mixture-of-Experts (SMoE).

The experts of our chosen models follow the style in LLaMA (Touvron et al., 2023), which consists of three linear layers and operates as:

$$\text{Expert}(x) = W_{\text{down}}(W_{\text{up}} x \odot \text{Act}(W_{\text{gate}} x)) \quad (1)$$

where $\odot$ denotes element-wise multiplication and the activation function is abbreviated as Act. Given

the three weight matrices $W_{\text{up}}, W_{\text{gate}} \in \mathbb{R}^{d_{\text{mid}} \times d_{\text{hid}}}$ and $W_{\text{down}} \in \mathbb{R}^{d_{\text{hid}} \times d_{\text{mid}}}$, we define a neuron as the combination of the row vectors $W_{\text{up}}[i, :]$ and $W_{\text{gate}}[i, :]$, along with the column vector $W_{\text{down}}[:, i]$. Thus, each expert contains d_{mid} neurons.

3 Overview

Our experiments are conducted on several open-source MoE models, namely Mixtral 8x7B, DeepSeekMoE, and Grok-1, which demonstrate impressive performance across various domains. To further study the similarities and differences between a standard transformer and an MoE model, we include Mistral 7B (Jiang et al., 2023) as one of our investigated models. Some basic information about these models and the abbreviations of the model names used throughout our paper are summarized in Tab 1. The analysis is divided into two sections, focusing on the model parameters (**static**) and the model behaviors with text inputted (**dynamic**). Unless otherwise stated, cosine similarity is employed for all experiments involving similarity measurement.

4 Analysis of Static Parameters

From a high-level perspective, the knowledge a model learns is encoded in its parameters. Hence, investigating weight matrices is a nartural way to examine a model. In this section, we study the correlation between parameters of: (i) MoE experts (and FFNs for Mistral), (ii) gate embeddings, which are two vital components of MoE.

4.1 Weight Matrices of Experts

MoE models replace FFNs in standard transformers with experts. Thus, following (Geva et al., 2020; Qiu et al., 2024), the projection matrices of the experts can be regarded as keys and values: the column vectors of W_{down} represent the possible outputs; the row vectors of W_{up} produce the weights for each possible output; the row vectors of W_{gate} decide whether to activate the corresponding neurons. Therefore, investigating the weight matrices is a straightforward way to understand the experts' behaviors. To study from different views, we analyze both the matrix and the neuron levels.

4.1.1 Matrix-level

In this part, we explore the similarity of weight matrices between all experts in every layer for the three projection matrices W_{up}, W_{gate}, and W_{down}.

Model	Abbreviation	# MoE layers	# experts	Topk	Hidden size (d_{hid})	Intermidate size (d_{mid})
Mixtral 8x7B	Mixtral	32	8	2	4096	14336
Mistral 7B	Mistral	32	N/A	N/A	4096	14336
DeepSeekMoE	DeepSeek	27	64 routed + 2 shared	6	2048	1408
Grok-1	Grok	64	8	2	6144	32768

Table 1: Basic information of models used for analysis. The abbreviations are used throughout our paper.

The similarity is calculated based on the flattened matrices vectors and illustrated in Fig. 1. We denote "F" as the Mistral FFN and "SE" as the DeepSeek shared expert in all the figures. Note that the figures of different models do not share the same color-bar.

Common[2]. The heat maps of the three matrices show similar patterns. Since the weight matrices are large, directly flattening them gives high-dimension vectors. We employ principal components analysis (PCA) to convert the flattened vectors to two-dimensional space. The resulting figures show that the expert distribution across the three weight matrices is also generally alike for Mixtral and DeepSeek. More details are presented in Append A.1.

Mixtral and Mistral. The cosine similarities between Mixtral experts (S_{ee}) mostly lie between 0.2 to 0.4, while the similarities between the experts and the Mistral FFN (S_{ef}) are about 0.6. Yet the values tend to be lower for deep (22th-30th) layers. A "dark cross" can be observed in some layers and corresponds with the outliers in the 2D space projected by PCA, indicating that the corresponding expert is relatively distinct from the others. Interestingly, this cross appears in Expert 3 the most frequently, which suggests that this expert might have learned some special attributes. It is worth noting that the cross usually pass through the entire heat map, including the last row of FFN. Hence, when a Mixtral expert differs from other experts, it is also less similar to the FFN of Mistral.

DeepSeek and Grok. Since the shared experts of DeepSeek are implemented as a single MLP block with hidden size larger than the routed experts, their flattened vectors cannot be directly compared and thus we omit the shared experts for this experiment. Fig. 1 demonstrates that the similarities between the DeepSeek routed and Grok experts are close to zero. While Mixtral's training method remains unrevealed, DeepSeek and Grok are known to be trained from scratch. This experiment shows that Mixtral might be trained by some special schemes, leading to less diverse experts training from scratch. (Wu et al., 2022)

---

[2]The observations shared by *all* of our investigated models are written in the Common part.

4.1.2 Neuron-level

In Sec 4.1, we measure the parameter similarity between experts in matrix level. However, the calculation of cosine similarity is position-related. Assume that the neurons of two experts are similar but in different order, the similarity of their weight matrices will be much lower than expected. To address this, we propose two approaches to investigate the correlation in neuron level: averaging and reordering. Averaging simply averages the rows (for W_{up} and W_{gate}) or the columns (for W_{down}) of the weight matrices and then calculates the similarity of the resulting vectors across experts. For reordering, we apply the Jonker-Volgenant algorithm, typically used for solving linear assignment problems, to find the optimal order of neurons to maximize the cosine similarity of the two experts. We describe the result of the reordering method below and leave the averaging details in Append B. Additionally, the projection of neurons in low-dimensional spaces using PCA can be found in Append A.2. Due to the heavy computation, we only pick several layers for reordering calculation. Note that the matrices are reordered separately. As described in Tab. 2, the order of Mixtral neurons hardly changes (large τ) and hence almost unchanged similarities. Despite the vast similarity growth for DeepSeek and Grok, their overall values are still about 1e-2.

Model	Order of Growth	τ
Mixtral	1e-3	0.75
DeepSeek	100	-0.0002
Grok	100	-0.0003

Table 2: Reordering results of expert neurons. τ denotes the Kendall's tau, whose value increases when two sequences exist strong agreement.

4.2 Gate Embedding

The gate in the chosen MoE models is implemented as a linear layer with embedding size $\mathbb{R}^{n_{exp} \times d_{hid}}$, where n_{exp} represents the number of experts. The gate is another main component of MoE besides the experts, and thus, it is crucial to study its attributes to better understand how MoE works. In addition, since each row vector in its embedding matrix determines the corresponding expert selection, some correspondence may exist between the

3
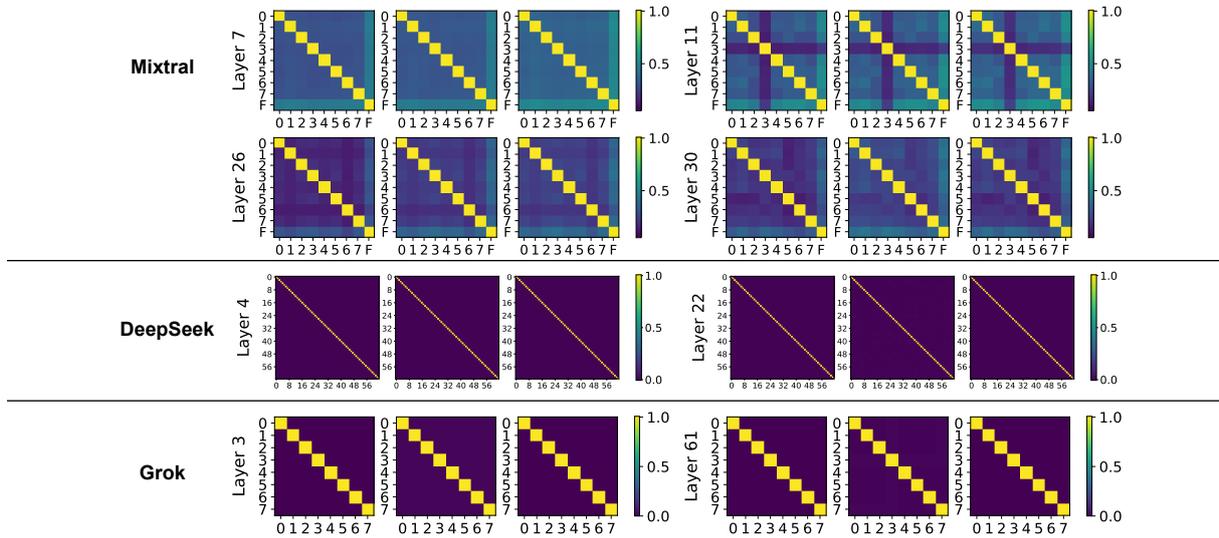
Figure 1: Matrix-level similarity heat maps of expert weight matrices. Each layer contains three heat maps, corresponding to W_{up}, W_{gate}, and W_{down}, respectively. The tick numbers refer to expert indices.

embedding matrix and the expert weights.

To investigate this, we measure the similarities between the gate embedding vectors. For computation simplicity, we compare them with the averaged heat maps (instead of the reordering results) in Append B, and the qualitative analyses are detailed in Append C. Specifically, we found that, for all three MoE models, the patterns in the heat maps of gate vectors and expert neurons in W_{gate} are partially alike in some layers (*i.e.*, the same coordinates in the two heat maps exhibit relatively higher/lower values simultaneously).

Therefore, we further conduct a quantitative analysis of their similarity values. In particular, we perform linear regression on the paired similarity dataset (X, Y), where X and Y denote the similarities of the gate vectors and one of the weight matrices W_{up}, W_{gate}, W_{down}, respectively. Tab 3 describes the square of Pearson correlation coefficients averaged over all layers (R^2_{avg}) and Tab 4 in the appendix lists the Pearson correlation coefficient (R) of every layer. As shown in Tab 3, the correlation between the similarities of the gate vectors and of W_{gate} is significantly stronger than W_{up} and W_{dwon}. For the (X, Y_{gate}) pair, although Mixtral and DeepSeek have similar R^2_{avg}, R^2 of Mixtral fluctuates between 0.1 and 0.7 while R^2 of DeepSeek stays near to 0.4. Furthermore, we can see from Tab 4 that (X, Y_{gate}) of both Mixtral and DeepSeek show positive correlations, whereas (X, Y_{gate}) of Grok turn to negative correlations starting from intermediate (>25th) layers. We note that the function of the gate embedding and W_{gate} is alike: the former determines the expert selection while the latter is responsible for choosing

neurons to activate. Therefore, they may learn similar knowledge to perform the *choosing* operation reasonably, hence the observed correlation.

Model	(X, Y_{up})	(X, Y_{gate})	(X, Y_{down})
Mixtral	0.06	0.33	0.07
DeepSeek	0.00	0.40	0.00
Grok	0.04	0.15	0.04

Table 3: Square of Pearson correlation coefficients averaged over all layers (R^2_{avg}) for three paired dataset.

4.3 Summary

Here, we conclude the inspiring observations in the analysis of static parameters: **(1)** Mixtral might contain expert(s) with special attributes. Dark crosses can be frequently found in Fig. 1. **(2)** The similarities of DeepSeek and Grok expert weight matrices are generally lower than those of Mixtral. As mentioned in Sec 4.1.1, the matrix-level similarities of DeepSeek and Grok experts are usually zero, whereas the Mixtral expert similarities reach about 0.3 on average. **(3)** Different experts' weights are less similar in deep layers. This can be observed from the Mixtral heat maps in Fig. 1. **(4)** W_{up}, W_{down}, and W_{gate}, share similar patterns in their similarity heat maps (Fig. 1). **(5)** The similarities of the gate embeddings and of W_{gate} show either positive or negative association. Tab 3 depicts the R^2_{avg} values, where the gate embeddings and W_{gate} pair achieve the highest for all three models.

5 Analysis of Dynamic Behaviours

The previous experiments examine the MoE models via their parameters, which do not involve any input. In this section, we feed text sequences into

4
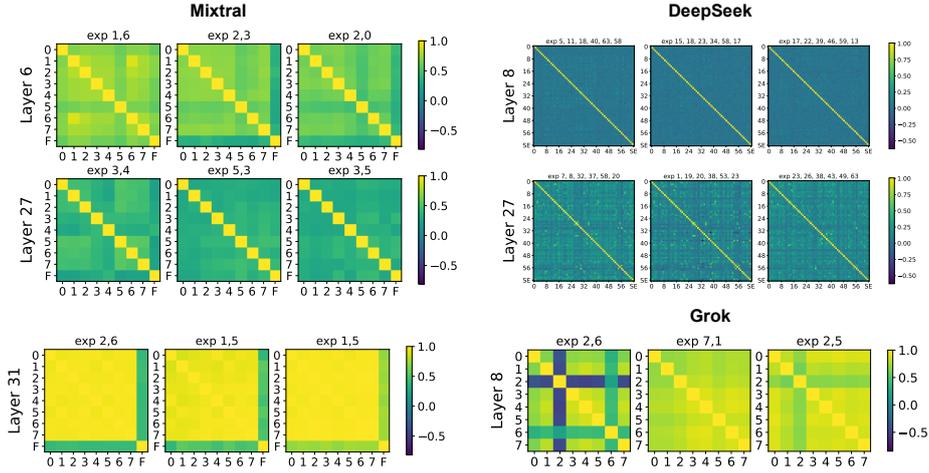
Figure 2: Similarity heat maps of expert output features using the short sequence. The top k experts for each token are shown on top of each heat map. Each number refers to an expert index.
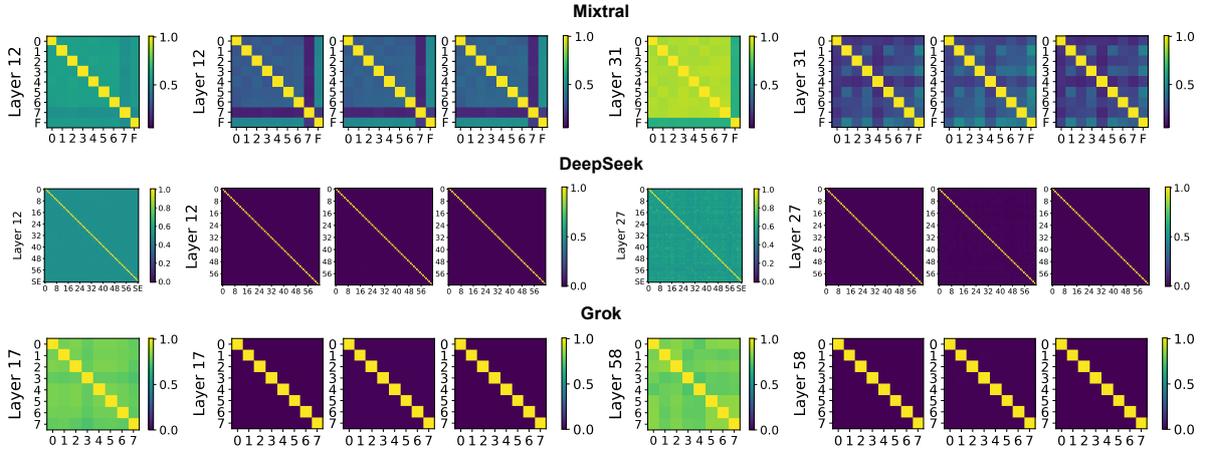
Figure 3: Averaged similarity heat maps of expert output features using the long sequence, plotted along with the matrix-level similarity heat maps. Each number refers to an expert index.

the MoE models to further study their actual behaviours given various inputs. Specifically, we analyze the outputs of the experts and gates.

To this end, two forward passes are needed. In the first pass, an input x is fed into the model for inference, and the output is stored for every layer. Then, the stored hidden state of the i-th layer is the input of the $(i + 1)$-th layer in the second pass, except for the first layer, which still employs x as the input. During the second pass, we modify the gate to route the input to all experts (*i.e.*, Topk=ALL) and record their outputs or intermediate states for analysis. Two different inputs are adopted, one contains 6 to 7 tokens[3] while another contains 1100 tokens (the number depends on the tokenizer).

5.1 Outputs of Experts

Since the experts, ideally, are learned to specialize in different aspects, it is natural to question the

---

[3]To be specific, the tokens are *<s>*, *As*, *an*, *open*, *source*, *alternative*, *to*, where the start of the sentence symbol *<s>* does not applicable for the Grok tokenizer.

similarities and differences between the outputs of selected and non-selected experts. In this experiment, we measure the correlation between the output feature vectors of experts using both input sequences. We plot the similarity heat maps for three tokens in the short sequence (Fig. 2) and the averaged heat map across all tokens in the long sequence (Fig. 3). Since the similarities have to be averaged for the long sequence, we employ angular similarity instead of cosine similarity for measurement so that the values range from 0 to 1:

$$\text{angular\_sim} = 1 - \frac{\arccos{(\text{cosine\_sim})}}{\pi}. \quad (2)$$

For observation convenience, the averaged similarity heat maps are plotted with the neuron-level similarity graphs of the expert weight matrices.

Mixtral and Mistral. The graphs of the short sequence show that the outputs from chosen experts tend to be more similar, which might be because their norms are generally larger. We will further discuss this in Sec 5.2. The overall similarities are rel-

5

atively low in the deep (22nd-27th) layers, whereas many values significantly grow to greater than 0.8 in the last two layers. Furthermore, having multiple dark crosses in a figure is common, while the experts corresponding to the dark crosses can be more similar to the Mistral FFN (*i.e.*, bright color in the last row). For the long sequence, the averaged heat maps have patterns similar to those of the neuron-level similarity graphs, including the dark crosses. The similarities also get lower with layer depth increasing, excepting the last layer. In addition, we have $S_{ee} > S_{ef}$ for both inputs. Most of the aforementioned observations are consistent with the previous analyses of static parameters (Sec 4.3), implying that measuring the similarity of weights, in some aspects, is equivalent to measuring the average similarity of outputs.

DeepSeek. Given the short input, most similarities are around zero, while the values in the last layer are significantly larger. Again, the similarities between experts chosen by the gate are likely to be higher, yet the difference emerges much less frequently than Mixtral. The averaged similarities of the long sequence also approach zero. Moreover, the amount of "small rectangular" with relatively light color in the graphs decreases when the layer gets deeper (except for the last layer), meaning that the averaged similarities are gradually lower.

Grok. Surprisingly, the similarities between the output features maintain a high level for all the tokens in the short sequence, indicating the experts are similar to each other in terms of behaviors. However, the similarities of their weight matrices are mostly zeros (Sec 4.1.1). We speculate that this is because the size of each Grok expert is relatively large. Each of them can learn comprehensive knowledge and behave alike despite distinct parameters. When averaging the similarities of the long input, some of the resulting averaged heat maps show similar patterns to the W_{gate} figures. This relationship aligns with Mixtral's observation.

5.2 Norms of Expert Outputs and Gate Scores

In Sec 5.1, we find that the outputs from chosen experts tend to be more alike. To investigate possible reasons for this observation, we study the relationship between the experts' L2 norm and the gate decision in this experiment. We employ the short sequence to be the input, and the calculated norms, along with the gate scores, are plotted in Fig. 4.

Mixtral. We found that the two experts chosen by the gate usually output feature vectors with the highest norms, which reveals that the norm might be one of the key factors for gate decisions. This finding agrees with the router's design in CompeteSMoE (Pham et al., 2024), which selects experts based on their output norms. It can also be regarded as an explanation for why the outputs of the chosen Mixtral and DeepSeek experts tend to be more alike (Sec 5.1). In Append D, we repeat this experiment using the long input, and the statistical results further demonstrate the "higher norm, higher score" observation. In Fig. 4, we can also see that the gate scores assigned for the top-1 experts are usually much higher than the others, including the second place. This demonstrates that the gate is learned to strengthen the confidence of decision during the training process. While the chosen experts output larger norms, the gate scores are not strictly proportional to the norms for the remaining experts. For instance, the expert with the lowest score might not output the smallest norm. On the other hand, the deeper the layer, the larger the norm, which is similar to the growth in standard models (Shleifer et al., 2021).

DeepSeek. In contrast to the observation about Mixtral experts, the gate decision seems to depend less obviously on the output norms of DeepSeek experts. However, the top 1 experts often score much higher than the remaining candidates. The magnitude of norms is proportional to the depth, yet the increment is smaller than Mixtral. In the last layer, the variance of norms becomes greater.

Grok. While the scores of the Top-1 experts are higher than the others, no correspondence between the norms and the gate scores is observed. One of the possible reasons might be the relatively low activation ratios of GeLU (see Sec 5.3) result in weaker dependence on the norm for the gate decisions. Besides, unlike Mixtral and DeepSeek, the magnitude of the norms hardly changes across the depth, and some of the norm values can be smaller than 1, which is rare in the other two models.

5.3 Intermediate States of Experts

While Sec 5.1 has studied the final outputs of experts, we continue to analyze their intermediate outputs here to examine the inner states of experts. Given an input x, the intermediate state of an expert refers to the output of $\text{act}(W_{\text{gate}}x)$, which is a d_{hid} dimensional vector. These intermediate vectors control the activation of neurons so we simply record them for analysis. The short sequence is used as the input. Mixtral, Mistral, and DeepSeek
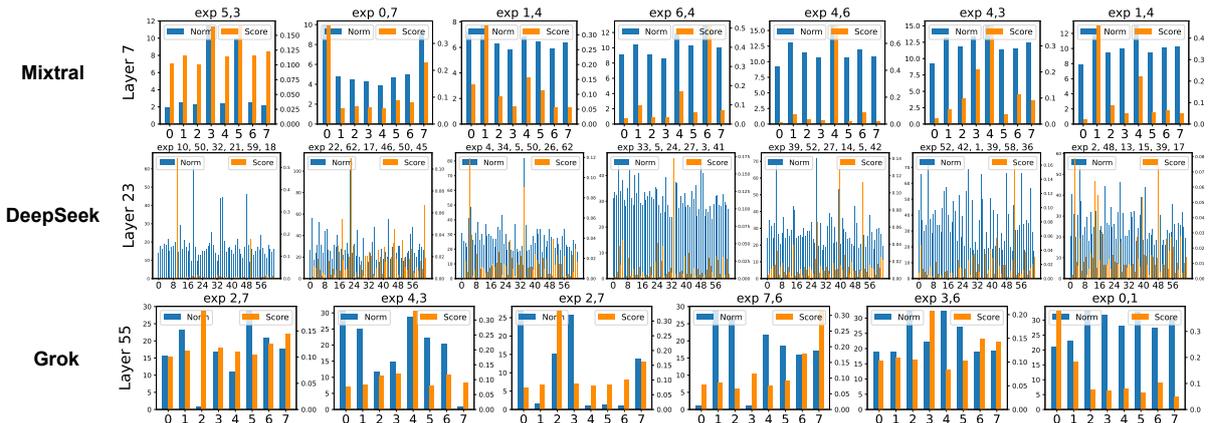
Figure 4: The experts' L2 norms and the gate scores of the short sequence. Each token's top k experts are shown on top of each heat map. Each number in the horizontal axis refers to an expert index.
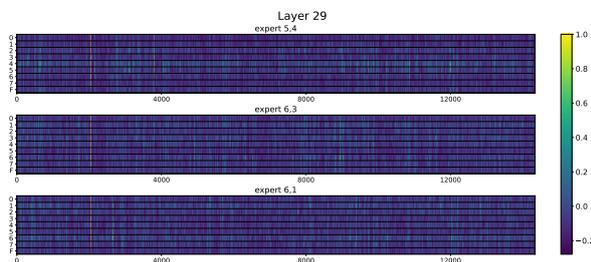
Figure 5: Intermediate state values of Mixtral experts. The top k experts are shown on top of each heat map. Each number in the vertical axis refers to an expert index while the horizontal axis represents the number of neurons.

utilize SiLU as the activation function while Grok adopts GeLU. Fig. 5 depicts the magnitude of the vectors. Due to the page limit, we only plot the results of Mixtral for three tokens.

Common. There are some horizontal lines in each figure, which means there is an 'out-liner' expert with the highest or lowest activation values. Nonetheless, there is no clear relation between such phenomena and gate decisions.

Mixtral and Mistral. For a single token, we found that the absolute activation value of 99.6% elements in each expert are greater than 0.001 after SiLU activation function on average. Such large ratio indicates that the vast majority of neurons in an expert are activated. In Fig. 5, some vertical lines across all experts are commonly found, meaning that the W_{gate} matrices of different experts assign similar activation values to neurons with the same indices. In addition, the magnitude of the intermediate states grows along with the layer depth, similar to the observation in Sec 5.1.

DeepSeek. On average, each expert's absolute

activation value of 99.7% neurons reaches beyond 0.001 after SiLU. Vertical lines rarely exist in the DeepSeek model. Similarly, the elements in the intermediate state vectors get larger as the layer goes deeper.

Grok. Employing GeLU as the activation function, only 25.3% neurons per Grok expert attain an absolute activation value greater than 0.001. The activation values are generally smaller than Mixtral and DeepSeek, Li et al. (2022); Song et al. (2024a) suggest such difference largely comes from different activation functions. Interestingly, Song et al. (2024b) further utilize the sparsity in experts in SMoE and achieve SOTA performance when activate the same number of parameters.

5.4 Summary

The observations are concluded below: **(1)** The outputs of Mixtral and DeepSeek experts in deep/last layers are less/much alike. This can be observed from the heat maps for both the short (Fig. 2) and long (Fig. 3) inputs. **(2)** The averaged heat maps of expert outputs have similar patterns to the neuron-level similarity graphs (Fig. 3), implying that measuring the similarity of weights, in some aspects, is equivalent to measuring the average similarity of outputs. **(3)** The outputs of Grok experts are highly similar (Fig. 2), which may be due to their large sizes. **(4)** For both Mixtral and DeepSeek, experts that output feature vectors with larger norms are likely to achieve higher gate scores, as shown in Fig. 4. We further verified this observation in Fig. 4 in the appendix. **(5)** For Mixtral, neurons of different experts positioned at the same indices have similar activation values. In Fig. 5, some vertical lines can be found in the Mixtral heat maps.

6 Discussion

Based on our analyses, we provide some suggestions for MoE models in various aspects:

- **Neuron-level experts**: Intuitively, the gate embedding determines the expert selection while W_{gate} is responsible for choosing neurons to activate. Meanwhile, we found that the similarities of the gate embedding and W_{gate} show association. This implies that the neurons might be some more fine-grained experts. Therefore, operations on experts, such as division, construction, and composition, should be further studied at the micro-level.

- **Model architecture**: Since the similarities between experts tend to be relatively low/high in deep/last layers, one can consider increasing the number of experts in deep layers while reducing it in the last layers. In addition, the gate is found to select experts with larger output norms frequently, so employing the norm as the routing mechanism is reasonable (Pham et al. (2024) have empirically proven the effectiveness).

- **Correlation measurement**: When analyzing the correlation between experts, measuring the similarities between their weight matrices gives partly equivalent results to measuring the similarities of their output feature vectors over considerable tokens. Hence, measuring the weight matrices can obtain an overview while inspecting the outputs of various tokens individually, which benefits fine-grained investigation.

- **Training scheme**: The Mixtral training method has not been publicly announced. However, we observed some attributes shared by Mixtral experts (*e.g.*, relatively high similarities of weight matrices), and certain relationship between experts and the FFN of Mistral (*e.g.*, similar intermediate states). Therefore, we conjecture that the Mixtral model is trained by some special initialization schemes other than from scratch, *e.g.*, upcycling (Komatsuzaki et al., 2022) from Mistral, that is, copying all experts from the FFN. On the contrary, the experts of DeepSeek and Grok, who are known to be trained from scratch, show a weaker correlation than Mixtral experts in our experiments. Hence, we speculate that compared with some initialization approaches, training a MoE model from scratch shows stronger potential to facilitate the diversification of experts.

7 Related Work

Most existing works analyze MoE from the router's perspective by observing expert selections. Early works have observed the unstable choices in the router (Zuo et al., 2021; Chi et al., 2022; Dai et al., 2022). Recent works find the standard router doesn't show clear specialization in the domain level (Jiang et al., 2024; Dai et al., 2024) and mainly routes based on token ID instead of high-level semantics (Xue et al., 2024). Some works investigate the expert's similarity (Wu et al., 2022), discovering and utilizing redundancies in experts for efficient inference (Li et al., 2023; Lu et al., 2024). Liu et al. (2023); Qiu et al. (2023) notice the connection between routing connection and expert computation and utilize the average of the experts' first layer weights to guide routing. Pham et al. (2024) proposes adding the expert's output norm as a supervision signal for routing training. These works provide insights into MoE from one or two viewpoints, while this work offers a systematic analysis and comparison.

8 Conclusion

In this paper, we initially attempt to investigate the inner working mechanisms of MoEs by studying the parameters and outputs of three different MoE models. We conclude the empirical observations and propose some practical suggestions for various aspects. It is still too early to answer whether MoEs do learn heterogeneous experts. Yet some of our experiments suggest that specific architecture designs (*e.g.*, the number of experts) and training frameworks might be more beneficial to expert specialization. We hope this work can provide some inspiring insights and serve as a valuable foundation for future research about MoE and other modular architectures.

9 Limitations

Our analysis is mainly based on observation and lacks intervention for further validation.

References

Zewen Chi, Li Dong, Shaohan Huang, Damai Dai, Shuming Ma, Barun Patra, Saksham Singhal, Payal Bajaj, Xia Song, Xian-Ling Mao, et al. 2022. On the representation collapse of sparse mixture of experts. *Advances in Neural Information Processing Systems*, 35:34600–34613.

Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y Wu, et al. 2024. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. *arXiv preprint arXiv:2401.06066*.

Damai Dai, Li Dong, Shuming Ma, Bo Zheng, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Stablemoe: Stable routing strategy for mixture of experts. *arXiv preprint arXiv:2204.08396*.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2020. Transformer feed-forward layers are key-value memories. *arXiv preprint arXiv:2012.14913*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Aran Komatsuzaki, Joan Puigcerver, James Lee-Thorp, Carlos Riquelme Ruiz, Basil Mustafa, Joshua Ainslie, Yi Tay, Mostafa Dehghani, and Neil Houlsby. 2022. Sparse upcycling: Training mixture-of-experts from dense checkpoints. *arXiv preprint arXiv:2212.05055*.

Pingzhi Li, Zhenyu Zhang, Prateek Yadav, Yi-Lin Sung, Yu Cheng, Mohit Bansal, and Tianlong Chen. 2023. Merge, then compress: Demystify efficient smoe with hints from its routing policy. *arXiv preprint arXiv:2310.01334*.

Zonglin Li, Chong You, Srinadh Bhojanapalli, Daliang Li, Ankit Singh Rawat, Sashank J Reddi, Ke Ye, Felix Chern, Felix Yu, Ruiqi Guo, et al. 2022. The lazy neuron phenomenon: On emergence of activation sparsity in transformers. *arXiv preprint arXiv:2210.06313*.

Zeyu Leo Liu, Tim Dettmers, Xi Victoria Lin, Veselin Stoyanov, and Xian Li. 2023. Towards a unified view of sparse feed-forward network in pretraining large language model. *arXiv preprint arXiv:2305.13999*.

Xudong Lu, Qi Liu, Yuhui Xu, Aojun Zhou, Siyuan Huang, Bo Zhang, Junchi Yan, and Hongsheng Li. 2024. Not all experts are equal: Efficient expert pruning and skipping for mixture-of-experts large language models. *arXiv preprint arXiv:2402.14800*.

Quang Pham, Giang Do, Huy Nguyen, TrungTin Nguyen, Chenghao Liu, Mina Sartipi, Binh T Nguyen, Savitha Ramasamy, Xiaoli Li, Steven Hoi, et al. 2024. Competesmoe–effective training of sparse mixture of experts via competition. *arXiv preprint arXiv:2402.02526*.

Zihan Qiu, Zeyu Huang, and Jie Fu. 2023. Emergent mixture-of-experts: Can dense pre-trained transformers benefit from emergent modular structures? *arXiv preprint arXiv:2310.10908*.

Zihan Qiu, Zeyu Huang, Youcheng Huang, and Jie Fu. 2024. Empirical study on updating key-value memories in transformer feed-forward layers. *arXiv preprint arXiv:2402.12233*.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Sam Shleifer, Jason Weston, and Myle Ott. 2021. Normformer: Improved transformer pretraining with extra normalization. *arXiv preprint arXiv:2110.09456*.

Chenyang Song, Xu Han, Zhengyan Zhang, Shengding Hu, Xiyu Shi, Kuai Li, Chen Chen, Zhiyuan Liu, Guangli Li, Tao Yang, et al. 2024a. Prosparse: Introducing and enhancing intrinsic activation sparsity within large language models. *arXiv preprint arXiv:2402.13516*.

Yixin Song, Haotong Xie, Zhengyan Zhang, Bo Wen, Li Ma, Zeyu Mi, and Haibo Chen. 2024b. Turbo sparse: Achieving llm sota performance with minimal activated parameters. *arXiv preprint arXiv:2406.05955*.

Qwen Team. 2024. Qwen1.5-moe: Matching 7b model performance with 1/3 activated parameters".

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Lemeng Wu, Mengchen Liu, Yinpeng Chen, Dongdong Chen, Xiyang Dai, and Lu Yuan. 2022. Residual mixture of experts. *arXiv preprint arXiv:2204.09636*.

Fuzhao Xue, Zian Zheng, Yao Fu, Jinjie Ni, Zangwei Zheng, Wangchunshu Zhou, and Yang You. 2024. Openmoe: An early effort on open mixture-of-experts language models. *arXiv preprint arXiv:2402.01739*.

Simiao Zuo, Xiaodong Liu, Jian Jiao, Young Jin Kim, Hany Hassan, Ruofei Zhang, Tuo Zhao, and Jianfeng Gao. 2021. Taming sparsely activated transformer with stochastic experts. *arXiv preprint arXiv:2110.04260*.

9

A Projection of Expert Matrices in Low-dimensional Space

A.1 Matrix-level

To better understand the expert relationship, we employ principal components analysis (PCA) to convert the flattened vectors of weight matrices to two-dimensional space. The vectors are standardized before applying PCA. Fig. 6 depicts the projection in 2D plane.

Mixtral and Mistral. Similar to the observation in Sec 4.1.1, the figures of the three matrices are alike. In general, about half of the Mixtral experts stay close to each other and to Mistral FFN, while the others locate much farther away. Moreover, the outliers correspond with the dark crosses.

DeepSeek. Only routed experts are considered due to different hidden sizes. Since several outliers exist and cause the remaining data points to be densely gathered, we remove them using the DBSCAN algorithm with $\epsilon = 50$ and plot the rest in Fig. 6. It can be observed that the experts distribute rather densely, especially for the up matrix. The distribution of experts varies for three matrices, but the figures of up- and down-matrices are more alike than those of the gate matrix.

Grok. Typically, about half of the Grok experts densely gather for the up and down matrices. Another half turns out to be outliers although no dark cross is observed before. Furthermore, the outliers of the three matrices are partly coincided.

A.2 Neuron-level

To project the neurons onto a 2D/3D space, each row vector of W_{up} and W_{gate}, or column vector of W_{down}, is treated as a single data point. Standardization is then applied, following by the PCA. Visualization of the principle components is illustrated in Fig. 7. Different colors refer to neurons of different experts.

Common. The vast majority of neurons gather in the low dimensional space. In some of the layers, the distribution of neurons forms a special shape, such as a cross or a thick line, which appears the most often for W_{down}, followed by W_{up} and finally W_{gate}. Comparing with ellipse, these shapes indicate that the neurons are relatively more similar.

Mixtral and Mistral. The neurons of Mistral FFN distribute more densely than those of Mixtral experts. Notably, the distribution shape of neurons in FFN and experts are usually alike, even for the outliers.

DeepSeek and Grok. The number of outliers is a bit greater than Mixtral.

B Averaging Expert Neurons

To investigate the expert correlation in neuron level, the averaging approach simply averages the rows (for W_{up} and W_{gate}) or the columns (for W_{down}) of the weight matrices and then calculates the similarity of the resulting vectors across experts. Fig. 8 displays the graphs.

Common. The heat maps of W_{up} and W_{down} are almost the same as in Section 4.1.1. Yet the similarities of W_{gate} significantly increase.

Mixtral and Mistral. The dark crosses sometimes disappear. In the figures of W_{gate}, the similarities between the experts and the Mistral FFN are often lower than the ones between the experts (*i.e.*, $S_{ee} > S_{ef}$), which is contrary to the previous observation. This can happen if the expert neurons in different positions are alike. For instance, given three vectors $f = (0,0)$, $e_1 = (1,0)$, and $e_2 = (0,1)$, the vector similarity $S_{e_1e_2}$ is lower than S_{e_1f} and S_{e_2f}. If averaging the elements, we have $\bar{f} = (0)$, $\bar{e}_1 = (0.5)$, and $\bar{e}_2 = (0.5)$, then $S_{e_1e_2}$ becomes the highest.

DeepSeek. The growth of the W_{gate} similarity values is directly proportional to the depth of layer.

Gork. In the heat map of W_{gate}, dark crosses are frequently appear in various positions.

C Gate Embedding

Since the output of each neuron determines the gate decision, there may exist some relationship between the embedding matrix of gate and of experts. To investigate this, we measure the similarities between the gate embedding vectors and compare with the averaged heat maps in Sec 4.1.2. The qualitative analyses for the combined graphs shown in Fig. 8 are detailed in this section. The table containing the R values of each layer (Tab 4) is also appended at last.

Mixtral. Focusing on the heat maps of gate embedding, the similarities typically range from 0.2 to 0.4, while the values in the last layer have a rather obvious gain. Moreover, dark cross is rarely found. Surprisingly, the patterns in the heat maps of gate embeddings and of expert neurons in W_{gate} are partially alike in some of the layers. This implies that the way a gate selects experts might be slightly relevant to the way an expert activates its neurons.
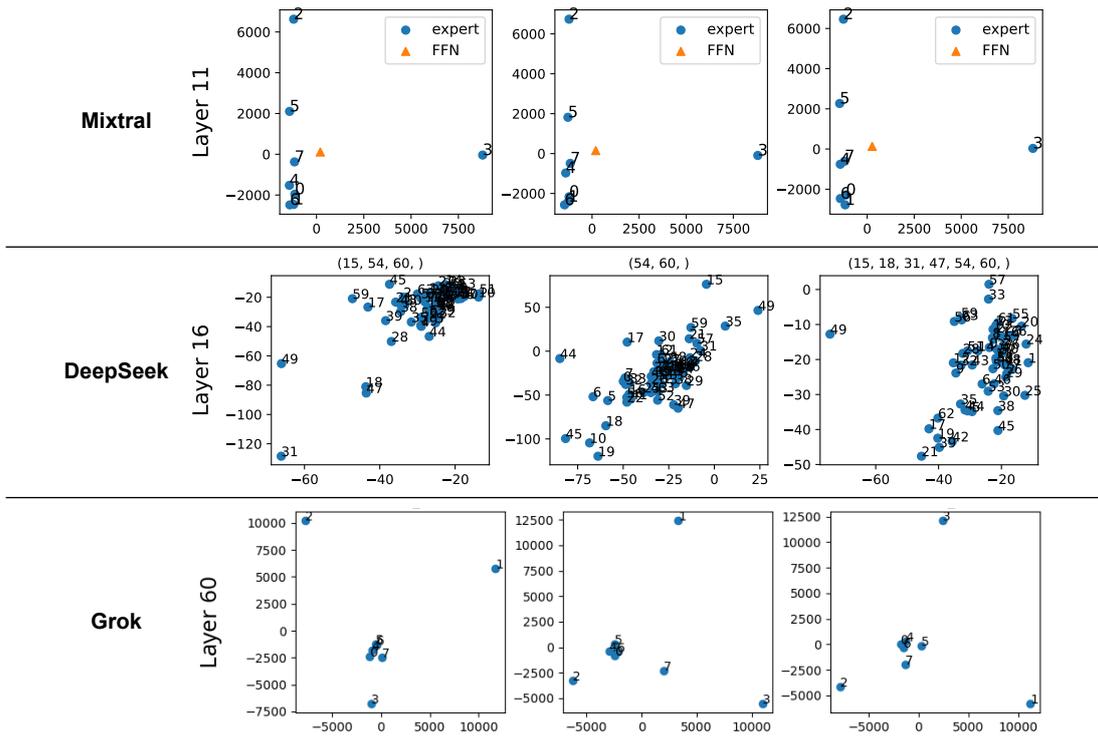
Figure 6: Projection of expert matrices in 2D space. Each layer contains three graphs, corresponding to W_{up}, W_{gate}, and W_{down}, respectively. For DeepSeek, the indices of the removed outliers are listed on top of each graph.
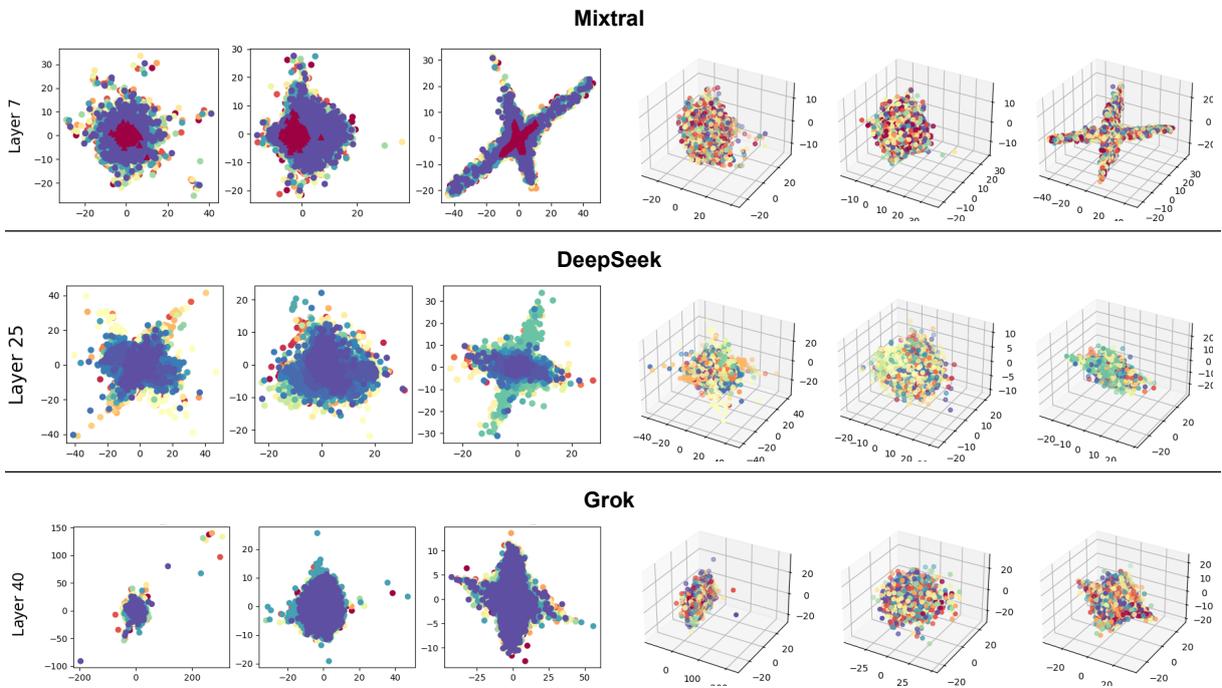
Figure 7: Projection of expert neurons in 2D/3D space. Each layer contains three graphs, corresponding to W_{up}, W_{gate}, and W_{down}, respectively.
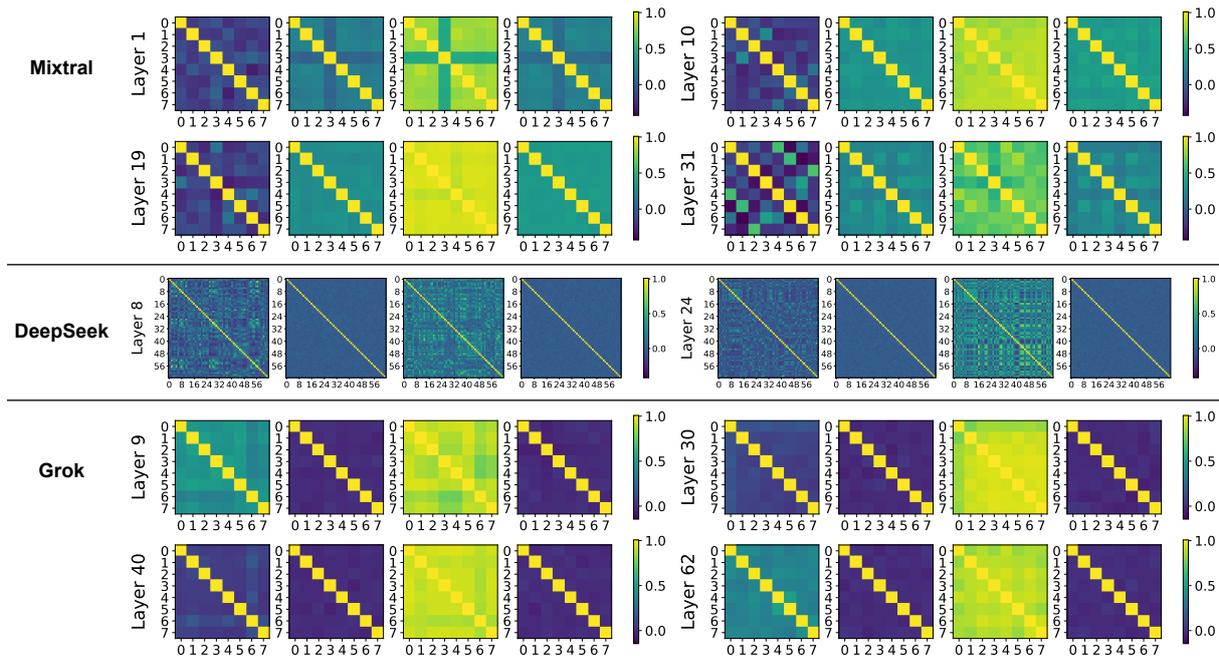
Figure 8: Similarity heat maps of gate embedding (leftmost graph of each layer) along with the neuron-level similarity heat maps using averaging method.

DeepSeek. Unlike the almost all-zero heat maps of W_{up} and W_{down}, the similarities of gate neurons sometimes get higher than 0.4. In addition, the heat maps of gate embeddings and of W_{gate} show similar patterns. However, the overall similarities of gate embeddings decrease with depth whereas the similarities of W_{gate} gradually grow. It means as layer goes deep, the gate "looks" at the input feature in more different ways when assigning scores to different experts, and meanwhile, the neuron activation of experts gets more alike.

Gork-1. Both dark and bright crosses commonly exist in the heat maps of gate embedding, whose patterns are similar to those of W_{gate}. Specially, their patterns show opposite color tendency (*i.e.*, the deep color positions in one heat map becomes light color in another) starting form the intermediate layers. The similarities of gate embedding get lower when the layer goes deep, except for the last few layers.

D Norms of Expert Outputs and Gate Scores

In Sec 5.2, we notice that in some MoE models, the two experts chosen by the gate usually output feature vectors with the highest norms. Therefore, we repeat the experiment using the long input and the statistical results are shown in Fig. 9.

Mixtral. It can be clearly observed that the ex-

pert which outputs the largest norm is assigned the highest score most often. Surprisingly, for every i, the i-th highest score is the most likely decision to be assigned to the expert with the i-th highest output.

DeepSeek. For the experts that output the first few largest norms (rank 60th to 64th), they are most likely to be assigned the highest scores. But we do not see similar relationship for the rest of the experts. On the contrary, the gate assigns relatively high scores more frequently than low scores to the experts with the smallest norms. For the experts whose output norms rank 49th to 59th, they are commonly assigned either low scores or high scores.

Grok. In contrast to the previous models, the output norms of the Grok experts tend to oppositely relate to the scores. More generally, the experts with the first few highest outputs are frequently assigned either low scores or high scores. One of the possible reasons might be the relatively low activation ratios of GeLU (see Sec 5.3) result in weaker dependence on the norm for the gate decisions.

E Chosen Experts

This experiment aims to examine the routing pattern. We feed an input prompt with about 64 tokens into the MoE models, and simply record the gate scores (softmax applied) of selected experts for
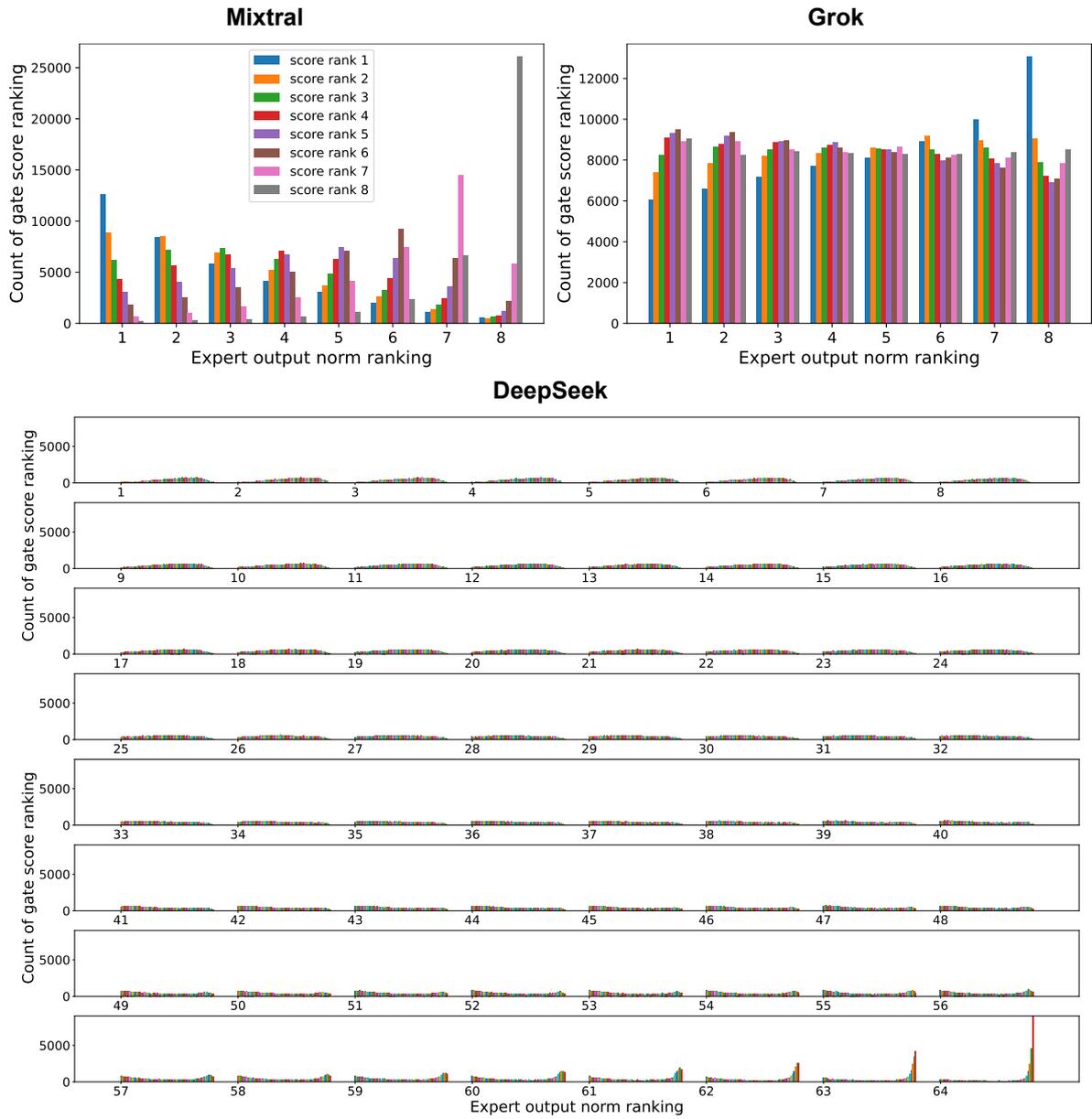
Figure 9: Counts of the gate score ranking for each norm ranking. The larger the rank number, the larger the norm.
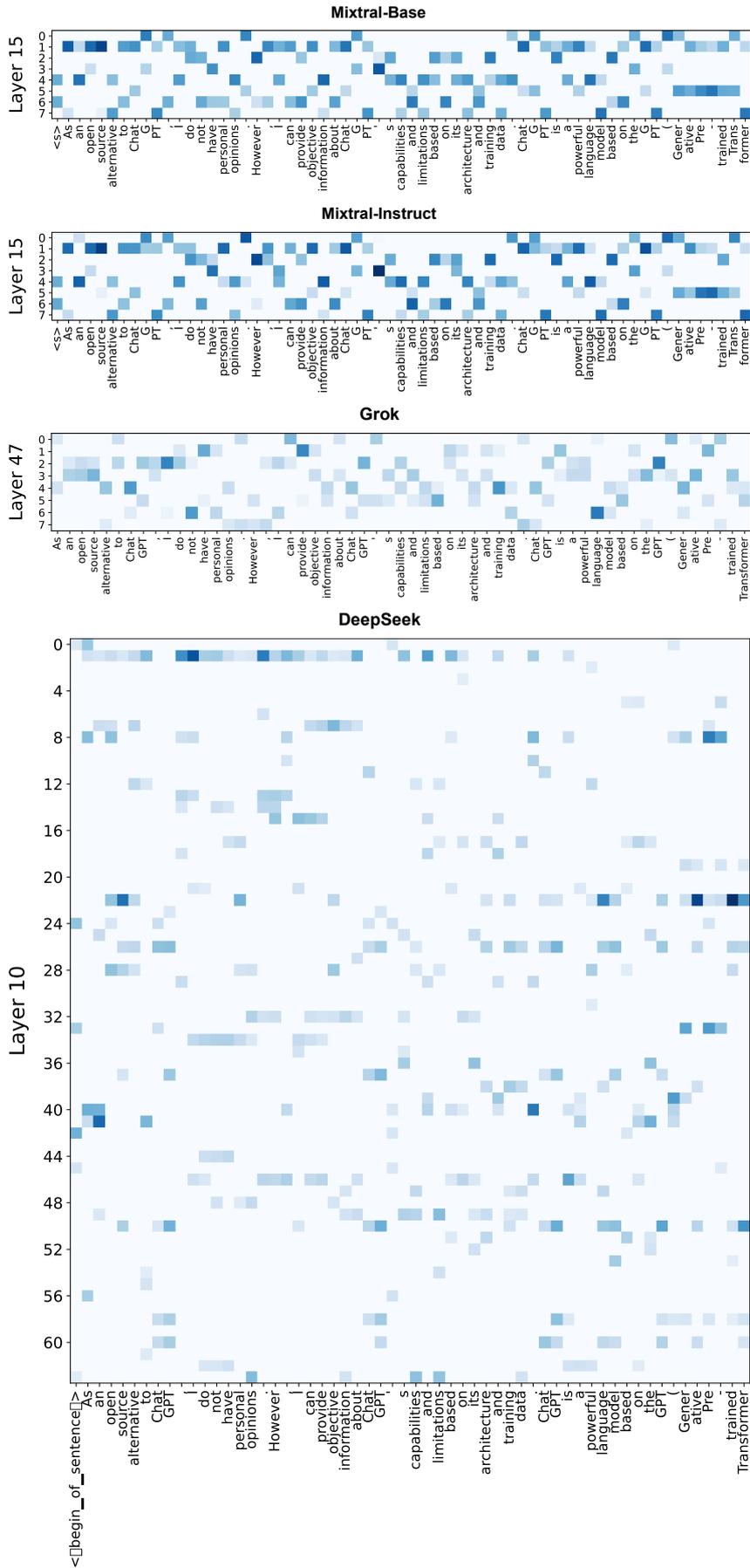
Figure 10: Routing pattern of different models. Deeper colors mean higher gate scores assigned to the corresponding experts. Only the scores of the top-k experts are illustrated.

Layer	Mixtral	DeepSeek	Grok
0	0.82	---	0.89
1	-0.44	0.75	-0.10
2	0.26	0.78	-0.28
3	0.54	0.71	0.66
4	0.48	0.77	0.52
5	0.70	0.77	0.37
6	0.84	0.69	0.28
7	0.74	0.73	0.17
8	0.42	0.66	0.51
9	0.66	0.66	0.84
10	0.53	0.63	0.28
11	0.32	0.60	0.30
12	0.14	0.54	0.46
13	0.51	0.60	0.14
14	0.66	0.56	0.00
15	0.40	0.58	0.54
16	0.39	0.53	0.32
17	0.53	0.55	0.30
18	0.35	0.57	0.10
19	0.17	0.57	-0.17
20	0.51	0.58	0.24
21	0.63	0.62	0.58
22	0.36	0.62	0.46
23	0.51	0.62	0.14
24	0.48	0.68	0.10
25	0.66	0.62	0.00
26	0.81	0.58	-0.10
27	0.63	0.46	-0.26
28	0.73	---	-0.66
29	0.75	---	-0.41
30	0.84	---	-0.83
31	0.57	---	-0.76
32	---	---	-0.24
33	---	---	-0.53
34	---	---	-0.46
35	---	---	0.14
36	---	---	0.17
37	---	---	-0.46
38	---	---	-0.17
39	---	---	-0.26
40	---	---	-0.70
41	---	---	0.17
42	---	---	0.00
43	---	---	-0.17
44	---	---	-0.22
45	---	---	0.14
46	---	---	-0.47
47	---	---	-0.44
48	---	---	-0.17
49	---	---	-0.14
50	---	---	0.17
51	---	---	0.22
52	---	---	0.10
53	---	---	0.33
54	---	---	-0.24
55	---	---	-0.57
56	---	---	-0.24
57	---	---	-0.37
58	---	---	0.00
59	---	---	-0.69
60	---	---	-0.17
61	---	---	0.35
62	---	---	0.30
63	---	---	0.10

Table 4: Pearson correlation coefficients (R) of the paired dataset (X, Y_{gate}).

879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904

each token. In addition to the base model of Mixtral (Mixtral-Base), we include its instruct version (Mixtral-Instruct) in this experiment. The results are depicted in Fig. 10.

Mixtral. The experts in Mixtral-Base are evenly selected across tokens and it is quite common to see a sequence with more than four tokens being routed to a same expert. But the "special expert" with the dark cross in previous similarity graphs turns out to be an exception. These special experts are less frequently chosen and are assigned relatively low scores. The routing pattern of Mixtral-Instruct is mostly identical to Mixtral-Base, which indicates fine-tuning has little impact on gate decisions.

DeepSeek. Some of the layers exist an expert that is selected by most of the tokens. However, no distinction is observed for these experts in previous similarity heat maps. Note that the gate scores of DeepSeek are typically lower than the scores of Mixtral because DeepSeek applies softmax before the top-k operation while Mixtral adopts the reverse way.

Grok. The expert selection is rather even and some relatively high scores exist in the deep (>30th) layers. Same as DeepSeek, the softmax is applied before the top-k operation for Grok.