MMNeuron: Discovering Neuron-Level Domain-Specific Interpretation in Multimodal Large Language Model

Anonymous ACL submission

Abstract

Projecting visual features into word embedding space has become a significant fusion strategy adopted by Multimodal Large Language Models (MLLMs). However, its internal mechanisms have yet to be explored. Inspired by multilingual research, we identify domain-specific neurons in multimodal large language models. Specifically, we investigate the distribution of domain-specific neurons and the mechanism of how MLLMs process features from diverse domains. Furthermore, we propose a three-stage framework for language model modules in MLLMs when handling projected image features, and verify this hypothesis using logit lens. Extensive experiments indicate that while current MLLMs exhibit Visual Question Answering (VQA) capability, they may not fully utilize domain-specific information. Manipulating domain-specific neurons properly will result in a 10% change of accuracy at most, shedding light on the development of cross-domain, all-encompassing MLLMs in the future. The source code is available at this URL.

1 Introduction

Neuron Analysis, which interprets activation of neurons as the recall of learned knowledge in deep neural networks, has been widely adopted by researchers to understand the inner workings of models (Sajjad et al., 2022; Fan et al., 2024). Prior studies have confirmed that certain neurons within deep neural networks play important roles in learning particular concepts (Oikarinen and Weng, 2022; Bai et al., 2024; Xiao et al., 2024), preserving factual knowledge (Chen et al., 2024; Dai et al., 2022; Niu et al., 2024) as well as solving specific tasks (Stanczak et al., 2022). Beyond enhancing model interpretability, current practical applications of Neuron Analysis include model distillation (Dalvi et al., 2020), knowledge editing (Chavhan et al., 2024; Pan et al., 2023), and controllable generation (Bau et al., 2019; Kojima et al., 2024). Central to such endeavors is the identification of neurons responsible for target scenarios.

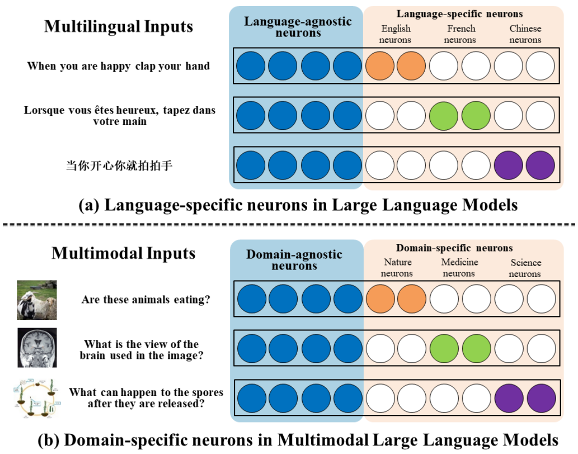


Figure 1: Neuron analysis in previous language-specific setting of large language model (a) and our domain-specific setting of multimodal large language model (b).

As illustrated in Figure 1 (a), recent studies have focused on interpreting the multilingual capabilities of pre-trained large language models (LLMs) under the view of *language-specific neurons*, which are neurons uniquely responsible for particular languages. For instance, Kojima et al. (2024) identified such neurons in pre-trained decoder-based language models, demonstrating that tampering with a few language-specific neurons significantly alters the occurrence probability of target language in text generation. Similarly, Zhao et al. (2024c) detected language-specific neurons by measuring the significance of neurons when processing multilingual inputs and proposed a workflow of LLMs handling multilingual tasks. Moreover, Tang et al. (2024) used language activation probability entropy (LAPE) to identify language-specific neurons, demonstrating that activating or deactivating certain neurons can change the language of the model's output. On the other hand, it has also been confirmed that neurons in text-only transformers

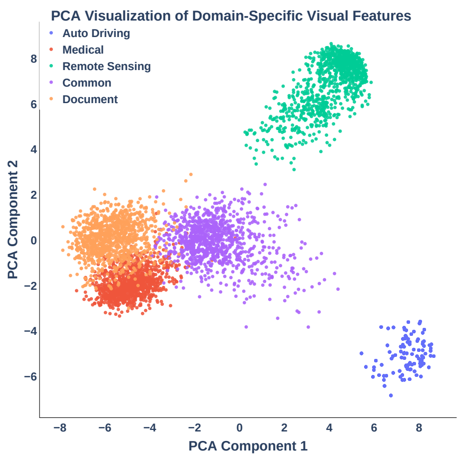


Figure 2: PCA visualization of image embeddings extracted through CLIP's image encoder.

can understand visual features extracted by a vision encoder (Schwettmann et al., 2023).

These findings have prompted an interesting question: *Do similar mechanisms exist in multimodal large language models (MLLMs) during the processing of features from different visual domains?* As shown in Figure 1(b), we aim to apply the mechanism similar to multilingual neuron analysis (Tang et al., 2024) to current representative open-source MLLMs, including LLaVA-NeXT (Liu et al., 2024a) and InstructBLIP (Dai et al., 2024). The aforementioned models extract image features via a pre-trained vision encoder and project these features into the word embedding space. These post-projection visual features are concatenated with language features and fed into the model's LLM module to generate text outputs.

Specifically, we investigate the activation patterns of neurons in MLLMs' feed-forward network (FFN) layers across corpora from five distinct domains, identifying less than 1% as domain-specific neurons. The datasets we utilized include LingoQA (Marcu et al., 2023), RS-VQA (HR) (Lobry et al., 2020), PMC-VQA (Zhang et al., 2023b), DocVQA (Mathew et al., 2021) and VQAv2 (Goyal et al., 2017), covering domains such as Auto Driving, Remote Sensing, Medicine, Document, and Common Scenes. Figure 2 highlights the clustering and separation of image features across the domains. Image examples of these domains can also be found in Appendix A. Based on our experiment results, we argue that differences exist among these visual domains and that the vision encoder and LLM modules in MLLMs exhibit distinct patterns for these domains. Furthermore, we propose a three-stage framework based on the distribution of domain-specific neurons among MLLM's LLM layers, where post-projection visual features are processed by LLM. To validate our hypothesis, we employ logit lens (nostalgebraist, 2020) to decode the hidden states of LLM's intermediate layers to visualize the feature transformation within transformer models (Vaswani et al., 2017).

Our main contributions are as follows:

- We identify the presence of domain-specific neurons in representative MLLMs, which is vital for interpreting domain-specific features.
- We analyze the impact of domain-specific neurons, indicating that both LLaVA-NeXT and InstructBLIP do not fully utilize domain-specific information in particular domains.
- We compare features from various domains through the lens of domain-specific neurons, revealing that images from different domains vary in conceptual depth.
- We propose a three-stage framework of language models in MLLMs when processing projected image features, shedding light on the internal mechanisms by which image features align with word embeddings.

To the best of our knowledge, we are the first to investigate domain-specific neurons in the multimodal field, although there are already insightful discussions on visual representations in MLLMs (Schwettmann et al., 2023; Zhao et al., 2024a). Our findings can reveal the neuron-level similarity and distinction among these domains, offering insights to understand and enhance the cross-domain potential of current MLLMs.

2 Related Work

2.1 Neuron Analysis

Neuron analysis has been recently widely explored in computer vision and natural language processing, which views neuron activation as the recall of learned knowledge (Mu and Andreas, 2020; Sajjad et al., 2022). Bau et al. (2017) propose to automatically inspect the functionality of each visual neuron in CNNs by evaluating the alignment between individual hidden units. Hernandez et al. (2021); Oikarinen and Weng (2022); Bai et al. (2024) further extend this method to open-ended by labeling hidden neurons in visual models with natural language descriptions. Neuron analysis has also been adopted to analyze language models, including the ability of sentiment analysis (Radford et al., 2017), machine translation (Mu et al., 2024), knowledge storing (Dai et al., 2022; Zhao et al., 2024b; Chen et al., 2024) and task solving (Wang et al., 2022). Recent research has associated specific neurons in LLMs with their multilingual ability, describing these neurons as language-specific neurons (Tang et al., 2024; Zhao et al., 2024c). Inspired by their work, we further expand this idea to the multimodal domain, being the first to analyze the domain-specific neurons in MLLMs.

2.2 Visual Representation in Word Embedding Space

Aligning image features within the word embedding space of LLMs has been one of the dominant frameworks adopted by current open-source MLLMs. Large Language and Vision Assistant (LLaVA) and its variants (Liu et al., 2024b, 2023a, 2024a) use a simple linear layer to connect image features extracted by the vision encoder of CLIP (Radford et al., 2021) into the word embedding space of LLMs (Touvron et al., 2023; Chiang et al., 2023; Jiang et al., 2023). Instead of concatenating post-projected embeddings directly with language instructions, InstructBLIP (Dai et al., 2024) employs a Q-Former to extract image features based on the instruction, which was more efficient. Similarly, MiniGPT-4 (Zhu et al., 2023) gained image features through pre-trained ViT (Dosovitskiy et al., 2020) or Q-Former (Li et al., 2022), which are then projected into the word space by a linear layer. Although such a framework has gained remarkable performance in various multimodal tasks (Antol et al., 2015; Chen et al., 2015; Liu et al., 2023b), the mechanism through which image tokens are processed by the LLM module still needed to be clarified. Our research has shed light on the interpretation of how MLLM understands the image tokens.

2.3 Cross-domain MLLM

Researchers have managed to fine-tune current general-domain MLLMs on specific domain corpus. For example, Kuckreja et al. (2024) train MLLM on the Remote Sensing multimodal dataset using LLaVA-1.5 architecture. LLaVA-Med (Li et al., 2023) was initialized with the general-domain LLaVA and then continuously trained in a curriculum learning fashion, while VLAAD (Park et al., 2024) opts for Video-LLaMA (Zhang et al., 2023a) as the foundational model to assist LLM in comprehending video data from auto driving scenarios. There are also researches trying to enhance MLLM's performance in specific domains (Bazi et al., 2024; Seyfioglu et al., 2023; Shao et al., 2023; Tian et al., 2024). Despite these efforts, it has also been proved that general-domain MLLMs without further domain-specific fine-tuning have demonstrated some cross-domain capability on some less common domains (Verma et al., 2024; Lu et al., 2023). In our research, we select virgin (i.e., without further fine-tuning) LLaVA-NeXT and InstructBLIP as our baseline, hoping to bring insights into the interpretation of general-domain MLLM's cross-domain potential and the development of all-around MLLMs qualified for different domains.

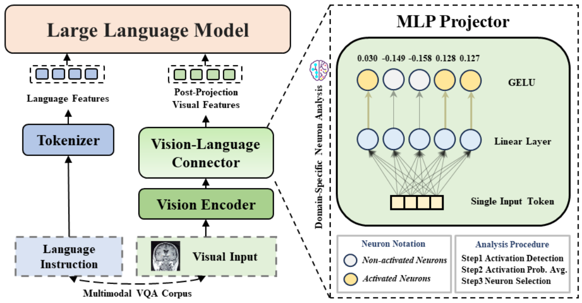


Figure 3: The overall framework of our proposed MM-Neuron method (taking LLaVA architecture as an example), which can be applied to any MLP layers with an activation layer in multimodal large language models.

3 Method

3.1 Neuron Activation Detection

A prevalent framework for vision-language models involves utilizing a pre-trained vision encoder to extract image features Z_v from image X_v. These features are then aligned with the word embedding space via a projection module, yielding post-projection features denoted as H_v. This process can be formalized as follows:

$$H_v = f_\Pi(Z_v), \quad \text{with} \quad Z_v = f_\Theta(X_v). \qquad (1)$$

Here, $f_\Pi(\cdot)$ and $f_\Theta(\cdot)$ represent the projection module parameterized by Π and the vision encoder parameterized by Θ. In LLaVA, the projection module is a simple linear layer, whereas in InstructBLIP, it is implemented via a Q-Former (Li et al., 2022). The post-projection features are then concatenated with language instruction embeddings H_q and fed into an LLM to generate text answer X_a:

$$X_a = f_\Phi([H_v, H_q]), \qquad (2)$$

where $f_\Phi(\cdot)$ refers to the language model parameterized by Φ.

For each Feed-Forward Network (FFN) layer, we consider every activation function as a neuron, as depicted in Figure 3. Given the hidden state $h^i \in \mathbb{R}^d$ of the input of the i-th FFN layer, the output of the FFN layer can be expressed as:

$$h^{i+1} = \mathrm{act\_fn}(h^i W_1^i) W_2^i, \tag{3}$$

where $\mathrm{act\_fn}(\cdot)$ denotes the activation function (e.g., GELU in Figure 3), and $W_1^i \in \mathbb{R}^{d\times s}$ and $W_2^i \in \mathbb{R}^{s\times d}$ represent the parameters of first Linear Layer and second Linear Layer. Here, s is the intermediate size of FFN layer. Therefore, there are s neurons in this FFN layer. Conventionally, the j-th neuron inside the i-th FFN layer is activated only if its respective activation value $act\_fn(h^i W_1^i)_j$ exceeds zero (Nair and Hinton, 2010).

3.2 Domain-Specific Neuron Selection

Our selection method is based on (Tang et al., 2024). For each domain $D_i, i = 1, 2, ..., k$, we feed its image-text corpus into MLLM, and record the activated frequency of each neuron u as well as the total token nums $N_{u,i}$[1]. The activation probability of a neuron u in domain D_i is denoted as:

$$p_{u,i} = \frac{M_{u,i}}{N_{u,i}}. \tag{4}$$

We denote the probability distribution of neuron u across all domains as P_u:

$$P_u = (p_{u,1}, p_{u,2}, ..., p_{u,k}). \tag{5}$$

The distribution can be normalized to a valid probability distribution through L1 normalization:

$$\begin{aligned} P'_u &= (p'_{u,1}, p'_{u,2}, ..., p'_{u,k}), \\ \text{where} \quad P'_{u,i} &= \frac{P_{u,i}}{\sum_{j=1}^{k} P_{u,j}}. \end{aligned} \tag{6}$$

Such a valid probability distribution allows us to calculate its corresponding entropy, termed domain activation probability entropy (DAPE):

$$DAPE_u = -\sum_{j=1}^{k} p_{u,j} \log p_{u,j}. \tag{7}$$

[1]Note that neurons in the vision encoder and language model may receive different numbers of tokens, since projected image features are concatenated with language embeddings before being fed into language model.

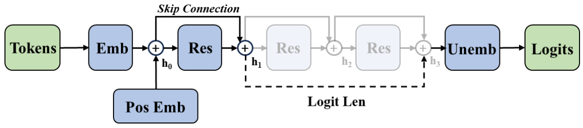


Figure 4: General Framework of logit len analysis, where it takes the hidden state at an intermediate layer (e.g., $h1$ above), and convert the hidden state into logits with the unembedding layer. Note that *Emb*, *Pos Emb*, *Res*, and *Unemb* stand for Embedding, Position Embedding, Residual Layer, and Unembedding, respectively.

Intuitively, a lower entropy indicates a tendency for activation in response to one or two domains, with reduced activation probabilities for others. Thus, neurons with low DAPE are designated as domain-specific neurons, following (Tang et al., 2024). In our work, we select those neurons with the bottom 1% DAPE scores as domain-specific neurons.

Upon identifying domain-specific neurons, we further analyze their specificity across five domains. A domain-specific neuron u is considered specific to domain D_j if its activation probability $p_{u,j}$ exceeds a predefined threshold.

3.3 Latent Embeddings Interpretation

Consider a transformer model, where its l-th layer updates the representation as follows:

$$h_{l+1} = h_l + F_l(h_l). \tag{8}$$

Here, F_i is the residual output of layer i. By applying Equation 8 recursively, the final output logits of model can be written as a function of an arbitrary hidden state h_i at the i-th layer:

$$logit(h_l) = LayerNorm(h_l + \sum_{i=l}^{L} F_i(h_i))W_U, \tag{9}$$

where the term $\sum_{i=l}^{L} F_i(h_i)$ represents the residual updates in the subsequent layers, and W_U denotes the so-called *unembedding matrix*. The *logit lens* approach involves setting the residuals to zero (Belrose et al., 2023):

$$LogitLens(h_l) = LayerNorm(h_l)W_U. \tag{10}$$

As shown in Figure 4, the logit lens decodes the hidden states of the transformer's intermediate layers into the distribution over the vocabulary, which can be used to interpret the model's latent embeddings (nostalgebraist, 2020). Ideally, the decoded distribution converges monotonically toward the next token predicted by the model.

We apply this trick to decode the hidden states of the language model, which allows us to understand the transformation of post-projection features within the language model module of the MLLM.

4 Experiment

In this section, we present empirical evaluation to elucidate the impact of domain-specific neurons, showing the potential mechanism of how MLLMs interpret image and language instructions.

4.1 Experimental Setup

4.1.1 Models

We study two public models: LLaVA-NeXT (Liu et al., 2024a) and InstructBLIP (Dai et al., 2024). The former utilizes a simple MLP layer to project image features extracted by CLIP's vision encoder into the word embedding space. The latter, however, employs the Q-Former (Li et al., 2022) to refine the image features extracted by ViT (Dosovitskiy et al., 2020). Specifically, we select llava-v1.6-vicuna-7b-hf[2] and Instructblip-vicuna-7b[3], both of which use Vicuna-7b (Chiang et al., 2023) as the language model base. The number of neurons in llava-v1.6-vicuna-7b-hf and Instructblip-vicuna-7b are 454.7K and 665.6K, respectively.

4.1.2 Dataset and Metrics

We select five datasets representing five different domains, namely, VQAv2 (Goyal et al., 2017) for common scenes, PMC-VQA (Zhang et al., 2023b) for Medical domain, DocVQA (Mathew et al., 2021) for Document domain, LingoQA (Marcu et al., 2023) for Auto Driving domain and RS-VQA (Lobry et al., 2020) for Remote Sensing domain. For LingoQA, visual instruction for each question includes multiple images, as shown in Figure 13a. We prepare image-question pairs of nearly the same token numbers for each domain during identifying, around 20 million tokens in LLaVA-NeXT. During evaluation, the scale of the validation set is aligned with LingoQA to make a fair comparison. For DocVQA, we report Average Normalized Levenshtein Similarity (ANLS) score (Biten et al., 2019) followed by the official benchmark. For LingoQA, we use the score of Lingo-Judge (Marcu et al., 2023) with the official implementation. For all other datasets, we report the top-1 accuracy (%) as the metric.

[2]https://huggingface.co/llava-hf/llava-v1.6-vicuna-7b-hf

[3]https://huggingface.co/Salesforce/instructblip-vicuna-7b

Baseline	Module	VQAv2	PMC-VQA	LingoQA	DocVQA	RS-VQA
LLaVA-NeXT	Vision Encoder	65	233	168	409	**465**
	MLP Projector	8	13	13	11	**20**
	LLM	683	915	1536	423	**2120**
InstructBLIP	Vision Encoder	94	488	279	**916**	891
	Q-Former	39	206	**334**	175	72
	LLM	410	774	**1567**	556	1419

Table 1: The number of neurons in each domain in different modules of MLLMs. **Bold** is used to highlight the domain with the most neurons in each module.

4.1.3 Implementation Details

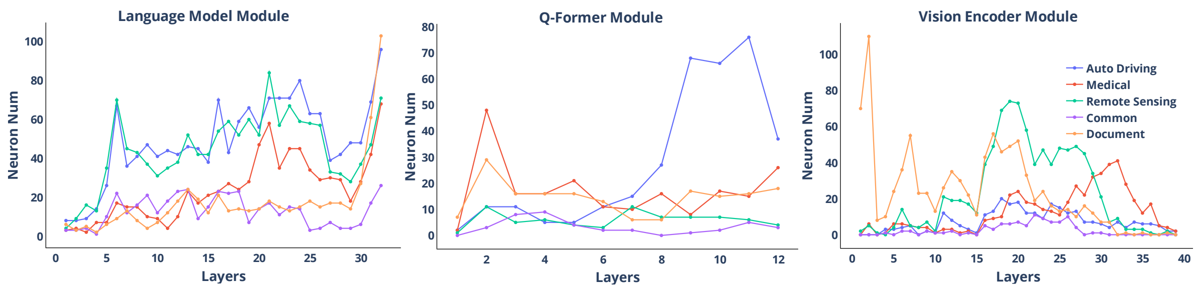
We adhere to the default prompt templates from the official repository or the original paper during evaluation, with an additional role description for the auto-driving scenes. For more details, please refer to Appendix B. We perform the forward pass without padding or truncation during the identification process. When evaluating models across different datasets, we employ beam search with *max_length* of 512 and *num_beams* of 5 to generate answers. The *temperature* and *length_penalty* arguments are set as 0.9 and -1, respectively.

4.2 Results & Discussion

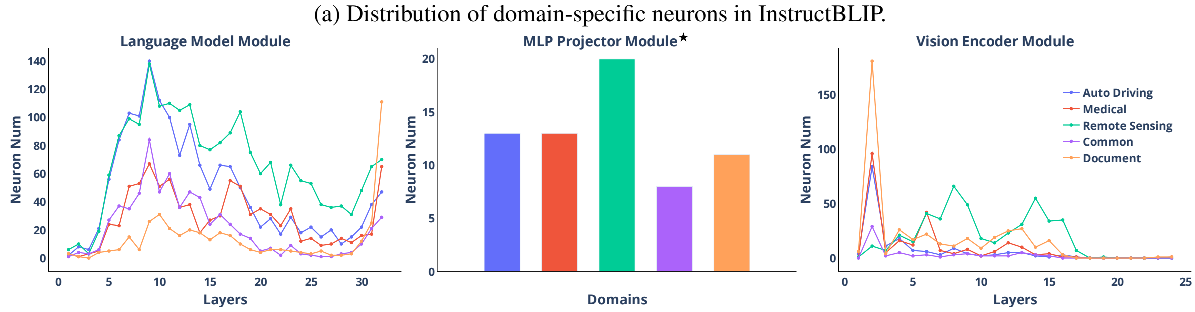
4.2.1 Distribution of Domain-specific Neurons

We identify domain-specific neurons using the method described in Section 3.2. Since neurons in different modules may have different activation patterns, as shown in Appendix C, we detected those domain-specific neurons module by module. Figure 5 shows the distribution of domain-specific neurons for each layer in each module of MLLMs.

Three-stage framework of LLM understanding multimodal features. Two obvious turning points can be observed in both LLaVA-NeXT and InstructBLIP's language model, one in the intermediate layer and the other near the output layer. Inspired by (Zhao et al., 2024c), we thus propose a three-stage framework of LLM understanding multimodal features: 1) In the first several layers, projected features are further aligned with word space. Around the turning point, the multimodal features are embedded into a uniform representation space, where included domain-specific information needs to be processed by more domain-specific neurons. 2) Transitioning into the second phase, features are further generalized and understood by language models, where domain-specific neurons decrease sharply. 3)In the third stage, language models gen-



(a) Distribution of domain-specific neurons in InstructBLIP.



(b) Distribution of domain-specific neurons in LLaVA-NeXT. $\star$: The MLP projector of LLaVA-NeXT consists of only one single layer.

Figure 5: Layer-wise Distribution of domain-specific neurons in different modules.

erate responses to the input, showing a rise of neurons specific to target tasks.

Our hypothesis aligns with the previous conclusion on smaller multimodal models like GPT-J (Wang and Komatsuzaki, 2021), as (Schwettmann et al., 2023) argue that outputs of the projection layer are further translated in deeper layers within the transformer. To further validate our hypothesis, we employ logit lens to visualize the transformation of multimodal features within language models in Section 4.2.3.

Domain-specific information in different semantic levels. Domain-specific neurons are mainly distributed in shallow and intermediate layers within MLLMs' vision encoders. Prior research discussed the correlation between the semantic level and layer depth, which found that more deep layers will focus on higher-level concepts in visual networks. In our settings, the document domain contains more low-level concepts, such as line and shape, while the remote sensing and medical domain may include more high-level concepts, like architectures and organs. Therefore, document neurons are mainly gathered in bottom layers close to the input end. Another interesting phenomenon is the rise of auto driving neurons near the output layer of InstructBLIP's Q-Former, we conjecture this may reflect the struggle of model to understand the language instructions of auto driving domain.

Gap between the ability of MLLM to handle visual and lingual instructions. Table 1 demonstrates the number of neurons in each domain. Remote sensing neurons have the largest proportion in LLaVA-NeXT's vision encoder, MLP projector and language model, while in InstructBLIP, the domain owns most specific neurons are document, auto driving and auto driving separately. We argue that the number of specific neurons reflects the understanding ability of MLLM in the target domain, as more specific neurons may mean more demanding to process domain-specific information. In contrast, less specific neurons mean more generalized features in the target domain (Tang et al., 2024). In this way, we find that there exists a large visual gap between domains like remote sensing, document and medical, comparing the two domains left. Moreover, InstructBLIP seems less proficient in processing questions from auto driving, as neurons of this domain exhibit the highest number in Q-Former and LLM. There is also a similar pattern in its language model as for the auto driving domain. In other words, while visual features of auto driving domain can be processed well by existing vision encoder, the language instruction of this domain may be hard to handle for language model.

4.2.2 Influence of domain-specific neurons

Perturbation for Performance in VQA Tasks Table 2 demonstrates the performance of LLaVA-

Model	Deactivated Module(s)	VQAv2	PMC-VQA	LingoQA	RS-VQA	DocVQA
LLaVA-NeXT	None	74.9	34.4	20.6	42.5	59.2
	Vision Encoder	75.8	**34.3**	24.6	42.1	58.3
	MLP Projector	74.9	34.4	24.2	42.5	59.2
	LLM	75.7	34.5	24.2	41.0	59.0
	All	**73.5**	34.5	**24.2**	**38.5**	**57.0**
InstructBLIP	None	66.1	28.1	20.6	34.7	24.0
	Vision Encoder	**66.9**	31.0	21.8	34.8	23.8
	Q-Former	67.1	32.4	20.0	**33.1**	24.6
	LLM	67.1	32.6	24.2	35.5	24.4
	All	68.6	**30.9**	**18.0**	33.6	**23.8**

Table 2: Accuracy (%) of LLaVA-NeXT and InstructBLIP on selected domains with corresponding domain-specific neurons deactivated. "None" means no neurons are deactivated, while "All" means deactivating domain-specific neurons in all the modules above. **Bold** is used to highlight the worst performance in each column.

Baseline	Module	VQAv2	PMC-VQA	LingoQA	DocVQA	RS-VQA
LLaVA-NeXT	Random (Avg.)	8.41	18.90	16.04	21.81	32.76
	LLM	0.01	0.01	0.02	0.10	0.02
	Vision Encoder	17.19	30.98	35.74	46.75	49.90
	MLP Projector	0.0	0.0	0.0	0.0	0.0
	All	**17.19**	**30.98**	**35.74**	**46.75**	**49.90**
InstructBLIP	Random (Avg.)	5.13	8.15	8.57	14.85	9.91
	LLM	6.84	12.13	9.62	7.80	11.98
	Vision Encoder	2.44	17.93	5.33	26.11	23.76
	Q-Former	2.93	11.61	6.95	14.58	6.52
	All	**8.00**	**24.84**	**12.77**	**29.04**	**26.58**

Table 3: The deviation (%) of hidden states of MLLMs' last layer after deactivating domain-specific neurons. We calculate the deviation $d=\frac{\|H_n-H_d\|_2}{\|H_n\|_2}$, where H_n and H_d denotes the hidden states before and after deactivating neurons separately. **Bold** is used to highlight the largest deviation in each column. *Random (Avg.)* refers to the average deviation by randomly deactivating neurons of the same number in all modules.

NeXT and InstructBLIP after deactivating domain-specific neurons in different modules. While the performance decrease after deactivating is slight for most domains, we find that deactivating remote sensing neurons in LLaVA-NeXT and auto driving neurons in InstructBLIP will result in a great fall of 4.0 and 2.6 accuracy separately. Similarly, in the document domain, deactivating domain-specific neurons at most causes a 2.2 accuracy decrease for LLaVA-NeXT. Interestingly, in some cases, removing domain-specific information seems to benefit the target task, as the accuracy of LLaVA-NeXT in auto driving has risen from 20.6 to 24.2. We leave this for future work.

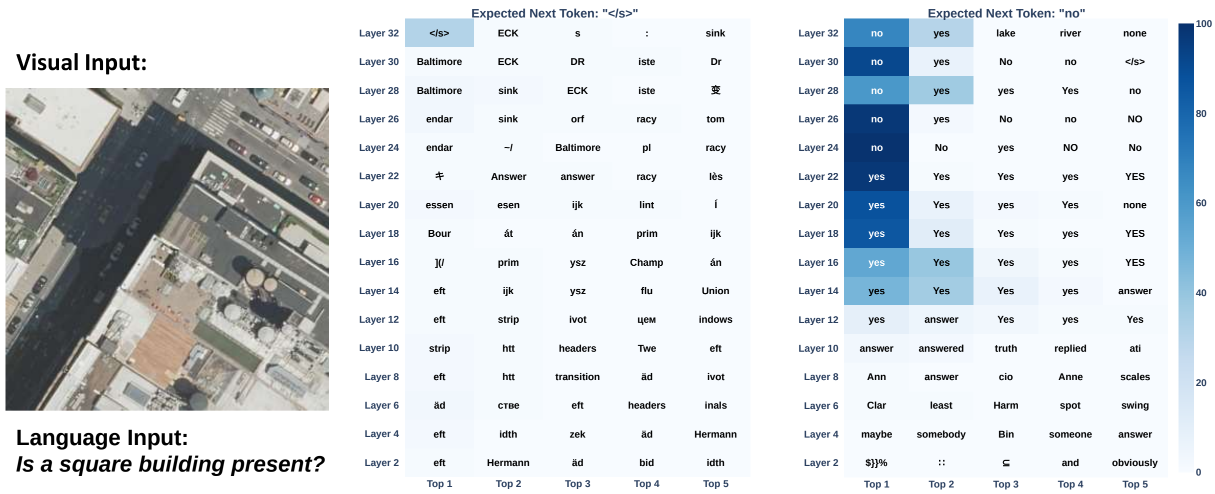
In summary, deactivating domain-specific neurons will not cause a sharp decrease in performance for some domains. To investigate the reason for that further, we compare the influence of domain-specific neurons in MLLMs' hidden states.

Perturbation for Hidden States We demonstrate the influence of domain-specific neurons on MLLM's last hidden states in Table 3. Surprisingly, deactivating domain-specific neurons causes a large perturbation to LLaVA-NeXT and InstructBLIP's hidden states. In contrast, deactivating all of the domain-specific neurons can have little effect on the accuracy of these domains, as shown in Table 2. Therefore, we argue that both LLaVA and InstructBLIP fail to take full advantage of the domain-specific information in specific domains, which may limit their cross-domain ability. In other words, the representations within MLLM's language models are highly generalized.

4.2.3 Case Study

To investigate how MLLM's language model processes image tokens, we employ logit lens (nostalgebraist, 2020) to decode the hidden states of the language model's intermediate layers into the probability of the next token across the vocabulary. As displayed in Figure 6, when feeding a remote sensing image-question pair into InstructBLIP, we get that the most likely token next to the second image token is '</s>', while the most likely token next of the last text token is the correct answer, 'no'. Interestingly, two place names, "Hermann" and "Baltimore", have appeared among the top token candidates when the image input is a remote sensing picture of New York. In multilingual literature, similar phenomena have also been observed. For instance, when Llama 2 receives the French token 'fleur' in the input, the English concept '__flower' will appear in the intermediate distribution (Wendler et al., 2024). This suggests that the decoded vocabulary distribution can to some extent reflect the semantic concepts understood by the language model. Despite this observation, we note that the decoded distribution of image tokens is far more sparse than text tokens; even in the output layer, the probability of the most likely next token '</s>' is lower than 40%. It indicates that projected tokens may be treated as a sparse mixture of concepts in the representation space instead of a simple word. We also demonstrate more cases of logit lens in different domains in Appendix D.

To further explore this phenomenon, we calculate the average entropy of the next token distribution for image tokens and text tokens separately, as shown in Figure 7. As the curves of image tokens tend to be above those of text tokens for all the layers, the next token distributions of image tokens are indeed more sparse than those of text tokens. Moreover, the tendency of entropy curves aligns with the hypothesis we have proposed in

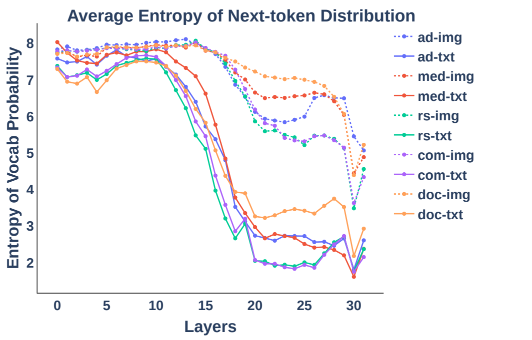


(a) Visaul and language input. The area in the image is located in New York.

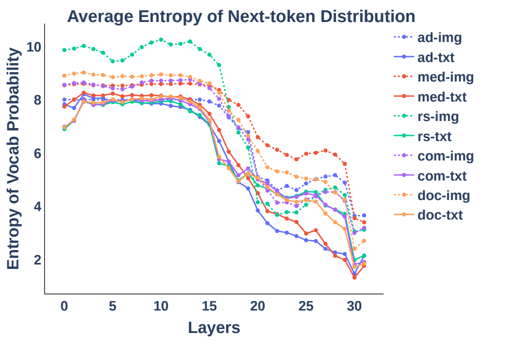
(b) The next token distribution of the second image token, the expected next token is '</s>' (i.e., end of sentence).

(c) The next token distribution of the last text token, the expected next token is the correct answer 'no'.

Figure 6: The logit lens can be applied to decode the hidden states of the language model's intermediate layers into the probability distribution of the vocabulary. We only display the top 5 candidates for each layer in the heatmap. Color indicates the probability of candidates from low (white) to high (blue).



(a) Average entropy of next-token distribution of InstructBLIP.



(b) Average entropy of next-token distribution of LLaVA-NeXT.

Figure 7: The average entropy of next token probability distribution for image and text tokens. The colors of lines denote different domains, such as auto driving (ad), remote sensing (rs), medical (med), common (com), and document (doc). We use dashed lines and solid lines to distinguish curves of image and text tokens.

Section 4.2.1. In the first stage, features are aligned into a uniform representation space, where entropy curves level off high. In the second stage, the language model understands and processes the information, as curves drop sharply in the intermediate layers. Finally, the model selects the suitable next token to output, resulting in a slight increase in entropy. A similar tendency has also been observed in English-native multilingual LLMs when handling non-English inputs (Wendler et al., 2024).

5 Conclusion

To explore the neuron-level domain-specific interpretation in current MLLMs, we propose MMNeuron framework inspired by multilingual research. In particular, we first calculate the activation probabilities of neurons in LLaVA-NeXT and InstructBLIP across five domains, identifying those with low domain DAPE scores as domain-specific neurons. By analyzing the distribution of domain-specific neurons and their influence on MLLMs, we find that the language model modules of MLLMs fail to fully utilize domain-specific information in VQA tasks. We further propose a three-stage framework that the language model module employs to handle projected visual features and corroborate it indirectly with logit lens. We envision that our work will shed light on the interpretability of current MLLMs, aiding the development of cross-domain, all-encompassing MLLMs in the future.

6 Limitations

Despite the findings we demonstrate in our work, there still exist several limitations:

1. Our experiments are conducted mainly on LLaVA-NeXT and InstructBLIP, whose frameworks are similar in aligning visual features with the word embedding space via a projector. This means that our findings may not be directly applicable to models that utilize different frameworks, such as those injecting vision representations into LLMs by layer (Wemm, 2023).

2. Although we find that domain-specific information is not fully utilized by the language model modules of MLLMs, how such information is conveyed and ignored between different layers is still less known. We leave these problems for future work.

3. We discuss the possible workflow of the language model module handling projected visual features through logit lens. While there do exist special semantic concepts in the decoded representations, we still know little about how these concepts are encoded and how projected features interact with word embeddings during the forward pass. Therefore, further mathematical analysis in this area is still required in the future.

References

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual Question Answering. In *International Conference on Computer Vision (ICCV)*.

Nicholas Bai, Rahul A Iyer, Tuomas Oikarinen, and Tsui-Wei Weng. 2024. Describe-and-dissect: Interpreting neurons in vision networks with language models. *arXiv preprint arXiv:2403.13771*.

Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2019. Identifying and controlling important neurons in neural machine translation. In *International Conference on Learning Representations*.

David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2017. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6541–6549.

Yakoub Bazi, Laila Bashmal, Mohamad Mahmoud Al Rahhal, Riccardo Ricci, and Farid Melgani. 2024. Rs-llava: A large vision-language model for joint captioning and question answering in remote sensing imagery. *Remote Sensing*, 16(9).

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*.

Ali Furkan Biten, Ruben Tito, Andres Mafla, Lluis Gomez, Marçal Rusinol, Ernest Valveny, CV Jawahar, and Dimosthenis Karatzas. 2019. Scene text visual question answering. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4291–4301.

Ruchika Chavhan, Da Li, and Timothy Hospedales. 2024. Conceptprune: Concept editing in diffusion models via skilled neuron pruning. *arXiv preprint arXiv:2405.19237*.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*.

Yuheng Chen, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. 2024. Journey to the center of the knowledge neurons: Discoveries of language-independent knowledge neurons and degenerate knowledge neurons. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17817–17825.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502, Dublin, Ireland. Association for Computational Linguistics.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. 2024. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36.

Fahim Dalvi, Hassan Sajjad, Nadir Durrani, and Yonatan Belinkov. 2020. Analyzing redundancy in pretrained transformer models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4908–4926, Online. Association for Computational Linguistics.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Yimin Fan, Fahim Dalvi, Nadir Durrani, and Hassan Sajjad. 2024. Evaluating neuron interpretation methods of nlp models. *Advances in Neural Information Processing Systems*, 36.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Evan Hernandez, Sarah Schwettmann, David Bau, Teona Bagashvili, Antonio Torralba, and Jacob Andreas. 2021. Natural language descriptions of deep visual features. In *International Conference on Learning Representations*.

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*.

Takeshi Kojima, Itsuki Okimura, Yusuke Iwasawa, Hitomi Yanaka, and Yutaka Matsuo. 2024. On the multilingual ability of decoder-based pre-trained language models: Finding and controlling language-specific neurons. *arXiv preprint arXiv:2404.02431*.

Kartik Kuckreja, Muhammad S. Danish, Muzammal Naseer, Abhijit Das, Salman Khan, and Fahad S. Khan. 2024. Geochat: Grounded large vision-language model for remote sensing. *The IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. 2023. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *arXiv preprint arXiv:2306.00890*.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2023a. Improved baselines with visual instruction tuning.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024a. Llava-next: Improved reasoning, ocr, and world knowledge.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024b. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. 2023b. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*.

Sylvain Lobry, Diego Marcos, Jesse Murray, and Devis Tuia. 2020. Rsvqa: Visual question answering for remote sensing data. *IEEE Transactions on Geoscience and Remote Sensing*, 58(12):8555–8566.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2023. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*.

Ana-Maria Marcu, Long Chen, Jan Hünermann, Alice Karnsund, Benoit Hanotte, Prajwal Chidananda, Saurabh Nair, Vijay Badrinarayanan, Alex Kendall, Jamie Shotton, and Oleg Sinavski. 2023. Lingoqa: Video question answering for autonomous driving. *arXiv preprint arXiv:2312.14115*.

Minesh Mathew, Dimosthenis Karatzas, and C.V. Jawahar. 2021. Docvqa: A dataset for vqa on document images. In *WACV*, pages 2200–2209.

Jesse Mu and Jacob Andreas. 2020. Compositional explanations of neurons. *Advances in Neural Information Processing Systems*, 33:17153–17163.

Yongyu Mu, Peinan Feng, Zhiquan Cao, Yuzhang Wu, Bei Li, Chenglong Wang, Tong Xiao, Kai Song, Tongran Liu, Chunliang Zhang, et al. 2024. Large language models are parallel multilingual learners. *arXiv preprint arXiv:2403.09073*.

Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814.

Jingcheng Niu, Andrew Liu, Zining Zhu, and Gerald Penn. 2024. What does the knowledge neuron thesis have to do with knowledge? *arXiv preprint arXiv:2405.02421*.

nostalgebraist. 2020. interpreting gpt: the logit lens. *LessWrong*.

Tuomas Oikarinen and Tsui-Wei Weng. 2022. Clip-dissect: Automatic description of neuron representations in deep vision networks. *arXiv preprint arXiv:2204.10965*.

Haowen Pan, Yixin Cao, Xiaozhi Wang, and Xun Yang. 2023. Finding and editing multi-modal neurons in pre-trained transformer. *arXiv preprint arXiv:2311.07470*.

SungYeon Park, MinJae Lee, JiHyuk Kang, Hahyeon Choi, Yoonah Park, Juhwan Cho, Adam Lee, and DongKyu Kim. 2024. Vlaad: Vision and language assistant for autonomous driving. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 980–987.

Alec Radford, Rafal Jozefowicz, and Ilya Sutskever. 2017. Learning to generate reviews and discovering sentiment. *arXiv preprint arXiv:1704.01444*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

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

Sarah Schwettmann, Neil Chowdhury, Samuel Klein, David Bau, and Antonio Torralba. 2023. Multimodal neurons in pretrained text-only transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2862–2867.

Mehmet Saygin Seyfioglu, Wisdom O. Ikezogwo, Fatemeh Ghezloo, Ranjay Krishna, and Linda Shapiro. 2023. Quilt-llava: Visual instruction tuning by extracting localized narratives from open-source histopathology videos. *Preprint*, arXiv:2312.04746.

Hao Shao, Yuxuan Hu, Letian Wang, Steven L Waslander, Yu Liu, and Hongsheng Li. 2023. Lmdrive: Closed-loop end-to-end driving with large language models. *arXiv preprint arXiv:2312.07488*.

Karolina Stanczak, Edoardo Ponti, Lucas Torroba Hennigen, Ryan Cotterell, and Isabelle Augenstein. 2022. Same neurons, different languages: Probing morphosyntax in multilingual pre-trained models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1589–1598, Seattle, United States. Association for Computational Linguistics.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024. Language-specific neurons: The key to multilingual capabilities in large language models. *arXiv preprint arXiv:2402.16438*.

Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Chenxu Hu, Yang Wang, Kun Zhan, Peng Jia, Xianpeng Lang, and Hang Zhao. 2024. Drivevlm: The convergence of autonomous driving and large vision-language models. *arXiv preprint arXiv:2402.12289*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Gaurav Verma, Minje Choi, Kartik Sharma, Jamelle Watson-Daniels, Sejoon Oh, and Srijan Kumar. 2024. Mysterious projections: Multimodal llms gain domain-specific visual capabilities without richer cross-modal projections. *arXiv preprint arXiv:2402.16832*.

Ben Wang and Aran Komatsuzaki. 2021. Gpt-j-6b: A 6 billion parameter autoregressive language model.

Xiaozhi Wang, Kaiyue Wen, Zhengyan Zhang, Lei Hou, Zhiyuan Liu, and Juanzi Li. 2022. Finding skill neurons in pre-trained transformer-based language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11132–11152, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Wemm. 2023. Wemm. `https://github.com/scenarios/WeMM`. Accessed: 2024-06-10.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in english? on the latent language of multilingual transformers. *arXiv preprint arXiv:2402.10588*.

Xiongye Xiao, Chenyu Zhou, Heng Ping, Defu Cao, Yaxing Li, Yizhuo Zhou, Shixuan Li, and Paul Bogdan. 2024. Exploring neuron interactions and emergence in llms: From the multifractal analysis perspective. *arXiv preprint arXiv:2402.09099*.

Hang Zhang, Xin Li, and Lidong Bing. 2023a. Video-LLaMA: An instruction-tuned audio-visual language model for video understanding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 543–553, Singapore. Association for Computational Linguistics.

Xiaoman Zhang, Chaoyi Wu, Ziheng Zhao, Weixiong Lin, Ya Zhang, Yanfeng Wang, and Weidi Xie. 2023b. Pmc-vqa: Visual instruction tuning for medical visual question answering. *arXiv preprint arXiv:2305.10415*.

Qinyu Zhao, Ming Xu, Kartik Gupta, Akshay Asthana, Liang Zheng, and Stephen Gould. 2024a. The first to know: How token distributions reveal hidden knowledge in large vision-language models? *arXiv preprint arXiv:2403.09037*.

Xin Zhao, Naoki Yoshinaga, and Daisuke Oba. 2024b. Tracing the roots of facts in multilingual language models: Independent, shared, and transferred knowledge. In *Proceedings of the 18th Conference of the*

European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2088–2102, St. Julian's, Malta. Association for Computational Linguistics.

Yiran Zhao, Wenxuan Zhang, Guizhen Chen, Kenji Kawaguchi, and Lidong Bing. 2024c. How do large language models handle multilingualism? *arXiv preprint arXiv:2402.18815*.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *Preprint*, arXiv:2304.10592.

A Visual Domain Definition

We define five domains in this work and each of them has characterized image features, as displayed in Table 4.

B Prompt Template

B.1 Instructions templates for VQA

For instructions with options, we separate options in alphabetical order, as shown in Appendix B.2. $\star$: A role description has been provided to help models better understand the tasks in auto driving. As shown below:

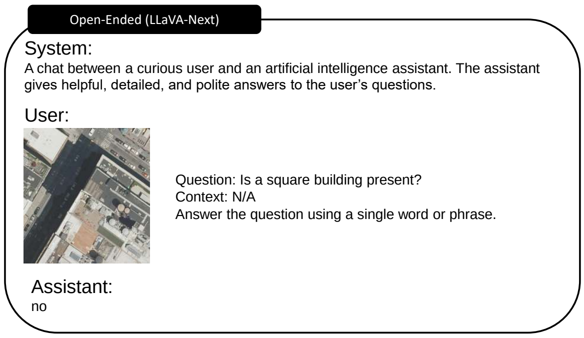
"Role: You are an advanced AI assistant installed on the Ego vehicle, equipped with conversational analysis capabilities for discussing autonomous driving scenarios. The perspective presented is from the point-of-view of the Ego vehicle, where the camera is mounted. It's important to note that the Ego vehicle itself is not visible in the images provided."

B.2 Prompt Examples

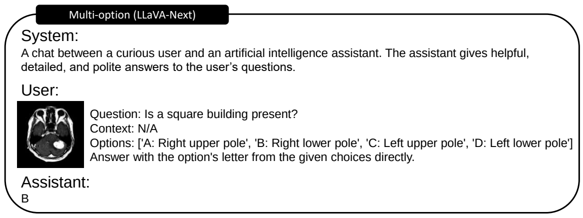
We display the prompt format we use for evaluation in LLaVA-NeXT, as shown in Figure 8. The prompt for InstructBLIP come from direct format in Table B.1.

C Silent Neurons in MLLM's Vision Encoder

We observed that several neurons in the vision encoders of LLaVA-NeXT and InstructBLIP remain silent regardless of the input images. We refer to these as "silent neurons". Figure 9 illustrates the distribution of these silent neurons within the vision encoders.

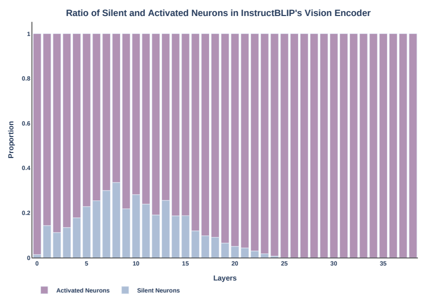


(a) Prompt example for open-ended tasks, the image and question come from RSVQA.

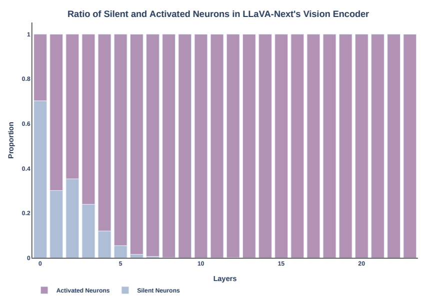


(b) Prompt example for multi-option tasks, the image and question come from PMC-VQA.

Figure 8: Prompt examples of conversational format for LLaVA-NeXT.



(a) Ratio of silent and activated neurons in IntructBLIP's vision encoder.



(b) Ratio of silent and activated neurons in LLaVA-NeXT's vision encoder.

Figure 9: Layer-wise distribution of silent neurons.

D Logit Lens Cases

We provide more cases from other four datasets, as displayed in Figure 10, 11, 12 and 13. For LingoQA (auto driving domain), the visual inputs for each question are multiple images.

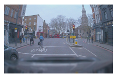
Domain	Definition	Dataset	Num of Samples	Example
Common Scenes	Natural images taken in everyday life	VQAv2 (Goyal et al., 2017)	21K	
Remote Sensing	Images captured by remote sensing sensors such as satellites	RS-VQA (Lobry et al., 2020)	11K	
Medical	Medical images obtained through techniques like CT and X-ray	PMC-VQA (Zhang et al., 2023b)	15K	
Document	Documents containing charts, text-rich images, and records	DocVQA (Mathew et al., 2021)	10K	
Auto Driving	Scenes captured from the viewpoint of a vehicle's camera	LingoQA (Marcu et al., 2023)	14K	

Table 4: Domain definition and the corresponding datasets.

Step	Model	Prompt
Identification	LLaVA-NeXT	<Image><Question>
	InstructBLIP	<Image><Question>
Evaluation (open-ended)	LLaVA-NeXT	A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions. **USER:** <Image> {Role Description}* Question: {Question} Context: N/A Answer the question using a single word or phrase. **ASSISTANT:**
	InstructBLIP	<Image> {Role Description}* Question: {Question} Short Answer:
Evaluation (multi-option)	LLaVA-NeXT	A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions. **USER:** <Image> Question: {Question} Context: N/A Options: {Options} Answer with the option's letter from the given choices directly. **ASSISTANT:**
	InstructBLIP	<Image> Question: {Question} Options: {Options} Answer with the option's letter from the given choices directly.

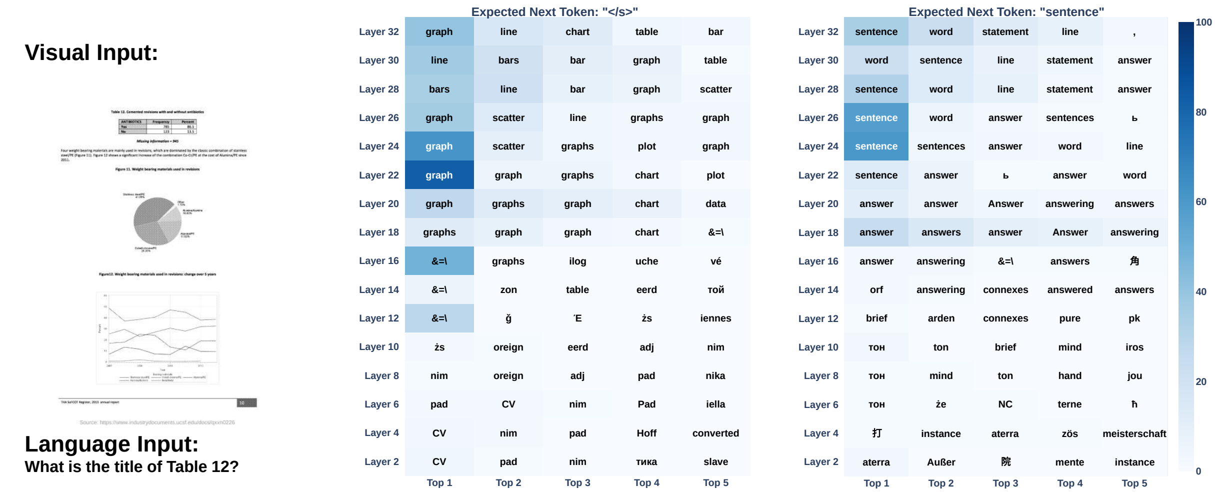
Table 5: Prompt templates we have used in different steps. For identifying domain-specific neurons, plain questions are input into models. During evaluation, we follow the templates provided by official repositories or codes.

(a) Visual and language input of PMC-VQA.

(b) The next token distribution of the 8th image token.

(c) The next token distribution of the last text token.

Figure 10: Case of logit lens in InstructBLIP on PMC-VQA.

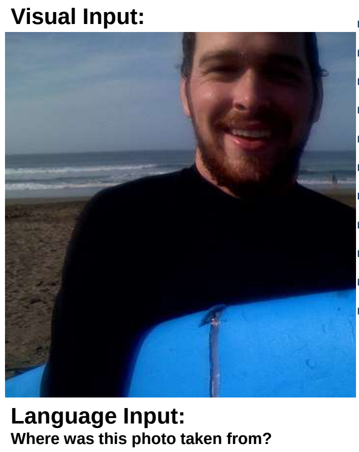


(a) Visual and language input of DocVQA.

(b) The next token distribution of the 377th image token.

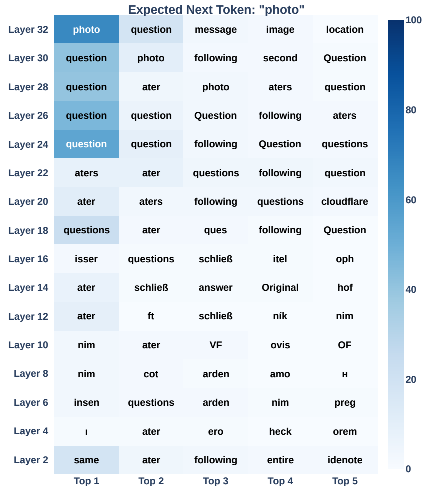
(c) The next token distribution of the 5th from last text token.

Figure 11: Case of logit lens in LLaVA-NeXT on DocVQA.



(a) Visual and language input of VQAv2.

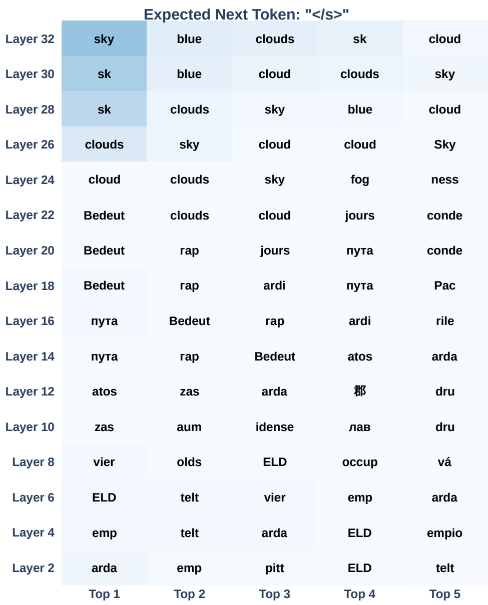
(b) The next token distribution of the 49th image token.



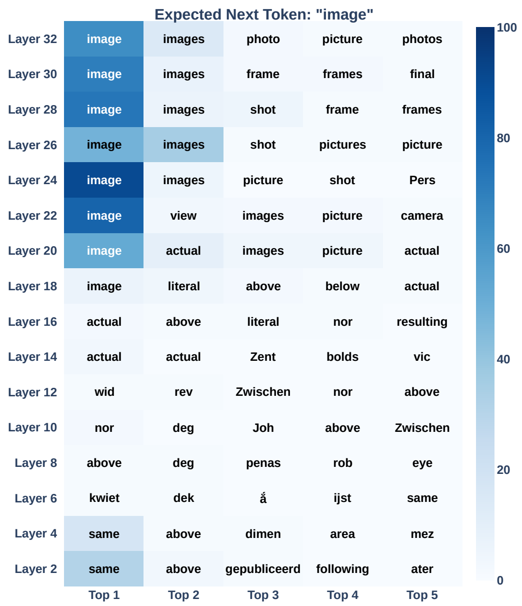
(c) The next token distribution of the 9th from last text token.

Figure 12: Case of logit lens in LLaVA-NeXT on VQAv2.

(a) Images inputs of LingoQA. Question: Is there a vehicle ahead of you in your lane?



(b) The next token distribution of the 37th image token in LLaVA-NeXT's vision encoder.



(c) The next token distribution of the 18th from the last text token.

Figure 13: Case of logit lens in LLaVA-NeXT on LingoQA.