Skipping Computations in Multimodal LLMs

Mustafa Shukor[1] * **Matthieu Cord**[1,2]

[1]Sorbonne University, [2]Valeo.ai

Abstract

Large Language Models (LLMs) have demonstrated remarkable success in both textual and multimodal domains. However, this success often comes with substantial computational costs, particularly when handling lengthy sequences of multimodal inputs. This has sparked many efforts focusing on enhancing efficiency during training and inference. In this study, we investigate the computation redundancy in Multimodal Large Language Models (MLLMs) during inference. We propose different methods to skip computations, such as skipping entire blocks, FFN or self-attention (SA) layers. Additionally, we explore parallelizing certain layers, such as FFN and SA layers. Our findings validate that (1) significant amount of computations can be avoided at inference time, especially for tasks such as Visual Question Answering (VQA). (2) Skipping computations during training can recover 97% of the original performance, even when skipping half of the blocks or removing 70% of the weights. Alternatively, (3) properly training with smaller LLMs can yield comparable performance to LLMs 2 or 3 times larger. To conclude, we extend our investigation to recent MLLMs, such as LLaVA-1.5, showing similar observations. Our work show that there is redundant computations inside MLLMs and thus the potential for significantly improving inference costs without sacrificing performance. The code is available here: https://github.com/mshukor/ima-lmms.

1 Introduction

Large Language Models (LLMs) [26, 82, 10, 52, 70] have been a major step towards human level intelligence. These models are capable of achieving reasonable scores on almost any textual task that can be done by humans.

Beyond the textual realm, LLMs are now the main building block for large multimodal models (LMMs) or multimodal LLMs (MLLMs) [2, 7, 14, 52]. However, training LLMs on more modalities requires significantly more computation resources due to the complexity of mulitmodal inputs. Multimodal inputs incur longer sequence length, additional encoders to tokenize different modalities and additional latency to preprocess each example.

Recent approaches have tried to overcome this burden by freezing all pretrained model parameters and training only the mapping module [48, 63, 47, 40, 11, 78, 71]. These models only train a modest number of parameters (few millions) and are able to attain reasonable performance on a wide range multimodal tasks. Besides the trainable parameters, complementary works tackle also data-efficiency and avoid costly multimodal pretraining [63, 47, 71]. When parameter and data-efficiency are combined, the training cost is reduced significantly, and becomes affordable by consumer grade GPUs.

However, reducing the inference cost of MLLMs is a problem that gather little of attention. These models, bottelnecked with an LLM, are significantly slow at inference and consume large amount of RAM and storage memory, hindering their deployment in the real world.

*Contact: {firstname.lastname}@sorbonne-universite.fr

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

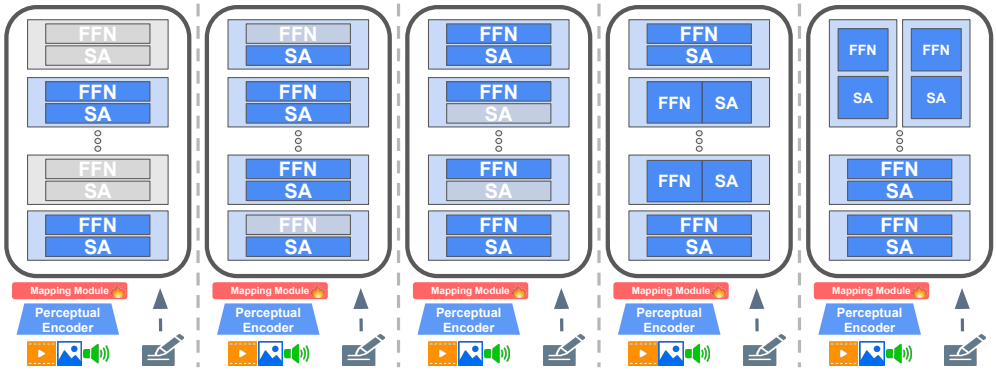

Figure 1: **Illustration of the proposed techniques to skip and parrallelize computations in multimodal LLMs.** From left to right: Skipping entire blocks (Skip Block), skipping only the FFN layers (Skip FFN), skipping the self-attention layers (Skip SA), parrallelizing the FFN and SA, parrellizing entire blocks. These are applied each interval I of layers, starting at a specific layer (sl).

In this work, we hypothesize that LLMs are highly overparametrized for general multimodal tasks, and contain redundant parameters, layers and blocks that can be bypassed.

Our work is also inspired by the recent study [61] that highlights the slowly changing embeddings for both textual and perceptual tokens, a phenomenon that has been observed with LLMs [67, 25, 41].

We investigate the computation redundancy at different granularity levels. We focus on skipping computations for both visual and textual tokens during autoregressive generation. Specifically, we start our study with pretrained models and test if we can skip entire blocks, FFN or SA layers and individual neurons, without any additional training. In addition, we also experiment with parrallizing the SA and FFN layers, or 2 entire blocks.

We continue our investigation focusing on skipping entire blocks, but during training. This helps to reduce both the training and inference cost. We show that we can retain more than 97% of the performance when training the mapping module with highly sparse LLMs; such as skipping half of the layers or removing more than 70% of the parameters. Finally, we show that almost the same original performance can be retained by properly training multimodel models with smaller LLMs.

In a nutshell, we propose a framework to study and compare different task-agnostic compression methods, for image, video and audio language tasks. Our study led to the following findings:

- Skipping computations for the generated textual tokens leads only to slight performance degradation, especially for VQA tasks.
- Training the mapping module while skipping computations can retain almost the original performance, even when 70% of the parameters are removed or 50% of the blocks are skipped.
- Properly training with smaller LLMs (*e.g.* OPT-2.7B) can achieve the same performance as larger ones (*e.g.* OPT-6.7B).
- Similar observations holds for the larger-scale MLLMs such as LLaVA-1.5.

2 Related work

LLMs and LMMs. Large Language Models (LLMs) [5, 26, 10, 56, 82, 70, 52] serve as the foundational components for contemporary Large Multimodal Models (LMMs). Most of these LMMs either utilize frozen LLMs [2, 3, 31] or fine-tune LLMs post-initialization [8, 7, 14]. Beyond scale, LLMs have facilitated the development of unified models capable of addressing multiple multimodal tasks [64, 49, 42, 72, 43, 12] in a unified framework. Efficient adaptation of unimodal models [48, 63, 78, 47, 81, 29, 84, 11, 51] represents a vital research avenue aiming to circumvent costly training by maintaining LLMs frozen and training a handful of adaptation parameters. Despite being significantly more efficient, they also compete with end-to-end trained models [35, 62, 13, 34, 65]

across image/video/audio and language tasks [63, 73, 75, 53]. However, despite their training efficiency, these models still incur significant costs at inference.

Compression for LLMs. Model compression has long been a research focus, particularly with the rise of large models. Approaches have been developed to effectively and scalably compress LLMs with hundreds of billions of parameters [5, 82, 56]. Post-training pruning methods aim to sparsify the model using a limited number of examples without additional model training. These methods employ layer-wise optimization [17, 18, 30, 19] or achieve efficiency with just a few model inferences [68]. While most pruning approaches target unstructured pruning, structured pruning results in actual wall clock time reduction [45, 41]. Post-training quantization [6, 58, 20, 15, 80] reduces the precision of model parameters to 8 bits, 4 bits, or lower, leading to increased efficiency and support across a wide range of hardware architectures. Early exiting methods directly generate the output from intermediate layers without using the last LLM layers [57, 9]. While conditional computations approaches, skip computations based on the input sample [1, 59, 55], our approach is static and input-agnostic.

Compression for multimodal models. While numerous approaches focus on compressing LLMs, fewer methods have been proposed for multimodal models. Existing approaches include knowledge distillation from powerful models [16, 74], unstructured pruning based on the Lottery Ticket Hypothesis (LTH) [22, 69], or structured pruning [60]. However, most of these methods require a significant amount of additional cost when applied to very large models such as MLLMs.

3 Framework for compressing preceptually augmented LLMs

3.1 General MLLMs framework

We investigate a general architecture comprising a frozen LLM, a trainable mapping module (C), and frozen perceptual encoders (E_M) for various modalities (M), such as image (I), video (V), and audio (A). The input to the LLM, denoted as X, is a concatenation of textual tokens ($T = [t_1, ..., t_{n_T}]$) and multimodal or perceptual tokens, referred to as the prompt ($P = [p_1, ..., p_{n_P}]$). The prompt is generated by encoding the modality-specific input (XM) with the corresponding encoder and projecting it to the LLM input space using the mapping module. This can be expressed as follows:

$$\begin{gathered} O = LLM\,(X) \\ X = [P;T] \quad P = C(E_M(XM)) \quad T = E_T(XT) \end{gathered} \tag{1}$$

The LLM consists of N blocks ${B^l}_{l\in\{0,...,N-1\}}$, where each block slightly refine the input tokens X^l:

$$X^N = \sum_{l=0}^{N-1} B^l(X^l), \tag{2}$$

And each block B can be expressed as:

$$\begin{aligned} X^{l+1} &= X_1 + FC2(g(FC1(LN2(X_1)))) \\ X_1 &= X^l + SA(LN1(X^l)), \end{aligned} \tag{3}$$

We mainly focus on the parameter/data-efficient setup, where both the LLMs and the perceptual encoders are kept frozen and only a light-weight mapping module (few millions parameters) is trained.

3.2 Efficient MLLMs baselines

We follow the same setup of previous works [63, 47, 37, 71], where we finetune one mapping module for each multimodal task. We train multiple models on image, video, and audio-text datasets. We adopt the baselines proposed in [61], that adopt a lightweight transformer with learnable queries

and self-attention mechanisms to attend to perceptual tokens. This transformer operates in a low-dimensional space, facilitated by down/up projection layers, and limits the number of learnable queries. Our baselines are similar to [71], however, we use significantly less number of learnable query (*e.g.*, 10) and prioritize a deeper architecture consisting of 5 blocks with hidden dimension of 256 over a wider one with a hidden dimension. We found this deeper architecture to work better in practice with reduced number of parameters. These baselines are trained with OPT-6.7B [82], and our study is complemented with Vicuna-v1.5-7B [83]. We utilize different powerful encoders for images (CLIP [54]), videos (X-CLIP [46]), and audios (AST [23]).

3.3 Implementation details

We adopt the implementation details outlined in [61]. Specifically, we employ the AdamW optimizer with a learning rate of 2e-4, which decreases using a cosine annealing scheduler to a minimum of 1e-5. During training, we use a total batch size of 16 for captioning and 64 for VQA datasets. We set the number of epochs to 20 to ensure convergence, although many models converge within just a couple of epochs. The best checkpoint is selected for evaluation; for instance, the image captioning model typically converges after approximately 4 epochs. Training is conducted on 8 V100 GPUs, and the duration varies depending on the task; for instance, each epoch takes about 30 minutes for the large VQAv2 dataset, while smaller datasets like Audiocaps and MSVD-QA require around 10 minutes per epoch.

3.4 Datasets and metrics

Following prior works, we select several public multimodal datasets that encompass two representative tasks: captioning and question-answering (QA) across various modalities, including image (VQAv2 [24], COCO caption [38]), video (MSVD [76], MSRVTT [77]), and audio (Audiocaps [28]). For QA datasets, we report accuracy in an open-ended generation setup with exact match, while for captioning, we report the CIDEr metric.

4 Skipping computations for MLLMs

4.1 Skipping computations

Method. We propose to skip entire layers in an input and task-agnostic manner. LLMs consists of many repetitive blocks which we argue that they are redundant and can be bypassed. In addition, previous works have demonstrated the slowly changing embeddings in LLMs [67, 25, 41] or MLLMs [61]. Specifically, when skipping entire blocks, Equation (2) can be written as:

$$X^{N-1} = \sum_{l \geq sl;\ l\ \%\ I=0;\ l \in \{0,\ldots,N-1\}} B^l(X^l), \tag{4}$$

Which means that the skipping happens starting from layer sl and each interval I (*e.g.*, $sl = 0$ and $I = 2$ skip half the blocks). Inside the block, we also investigate if we can skip FFN or SA layers. To skip FFNs, each interval (I) of layers, Equation (3) can be written as:

$$X^{l+1} = X^l + SA(LN1(X^l)), \tag{5}$$

And similarly when skipping SA layers:

$$X^{l+1} = X^l + FC2(g(FC1(LN2(LN1(X^l))))) \tag{6}$$

Experimental results. Fig. 2 presents a comparison of skipping blocks, feed-forward networks (FFNs), or self-attention (SA) layers across various multimodal datasets. For question-answering (QA) tasks, we observe that we can skip up to 33% of the blocks while retaining over 90% of the original performance. However, captioning tasks pose greater challenges due to the larger number of generated textual tokens, with the ability to skip between 15% and 25% of the blocks depending on the dataset. In general, skipping entire blocks yields the best results, whereas skipping SA layers results in the lowest performance, underscoring the significance of SA layers for these models.

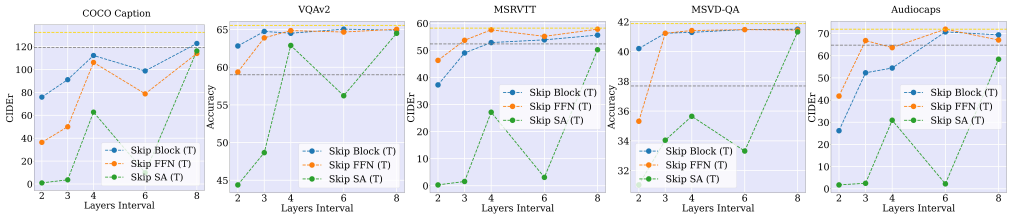

Figure 2: **Skipping computations inside MLLMs.** We skip entire blocks (Skip Block), FFN (Skip FFN) or SA layers (Skip SA). The skipping start at layer 4 and happen each couple of layers (Layer Interval). The gray line indicate 90% of original performance (shown in yellow).

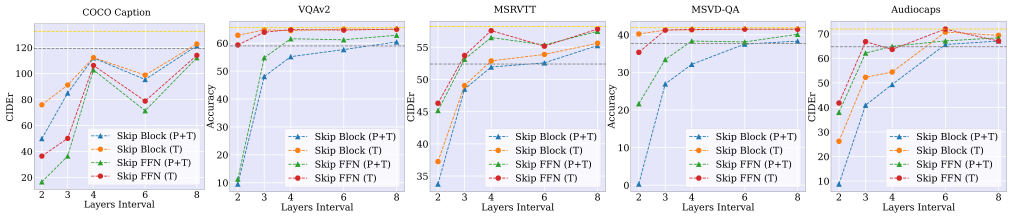

Figure 3: **Which tokens to skip?** We compare between skipping layers, only for the generated textual tokens (T), and all tokens including the prompts (P+T).

Which tokens to skip? In this comparison, we examine the impact of skipping only the generated textual tokens, as done in the previous section, versus skipping all tokens, including the prompt (P) containing perceptual tokens, the BOS token, and the textual tokens corresponding to questions in QA tasks. As illustrated in Fig. 3, for QA tasks, we observe that skipping up to 25% of the blocks for all tokens maintains 90% of the performance. Generally, skipping layers for the generated tokens yields higher scores.

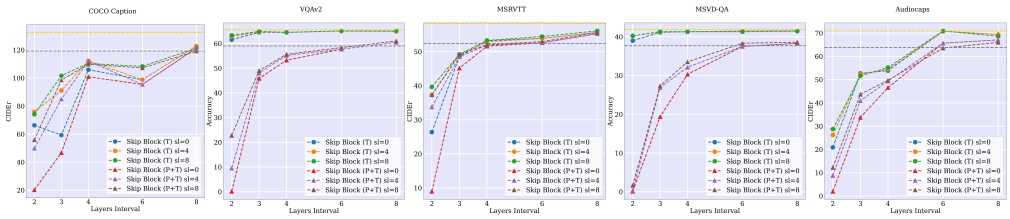

Figure 4: **Where to start skipping layers?** Skipping early layers leads to further decrease in scores. Starting at layer 8 (sl=8) leads to the best performance, especially when skipping many blocks.

Where to start skipping layers? Prior studies [61] have highlighted a significant change in both textual and perceptual embeddings at early layers, highlighting their importance, compared to later ones. To further investigate this phenomenon, we compare different starting layers for block skipping. As depicted in Fig. 4, we observe that avoiding skipping early layers leads to improvements, particularly when skipping a large number of layers (50%). This effect is more pronounced when skipping perceptual tokens as well.

4.2 Parrallelizing computations

Method. In this section, we propose to parrallelize different layers, this theoretically helps to reduce the inference time on GPUs by avoiding sequential computations. Specifically, we parrallelize entire blocks by replacing Equation (2) as follows:

$$X^{N-1} = \sum_{l \geq sl;\ l\ \%\ I=0;\ l \in \{0,\ldots,N-2\}} B^l(X^l) + B^{l+1}(X^{l+1}), \tag{7}$$

Inside the block, we also parrallelize the FFN and SA each interval (I) of layers by expressing Equation (3) as follows:

$$X^{l+1} = X^l + FC2(g(FC1(LN2(X^l)))) + SA(LN1(X^l)) \tag{8}$$

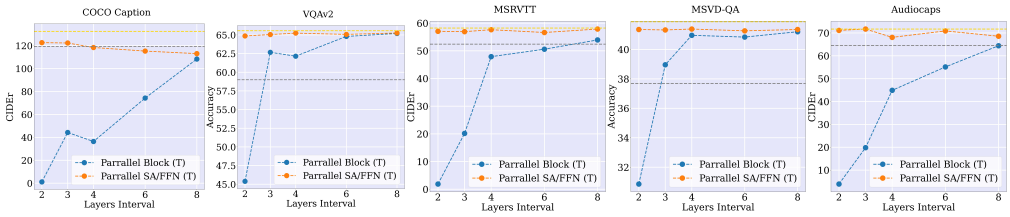

Figure 5: **Parrellelizing computations inside MLLMs.** FFN and SA layers can be cast in parrallel instead of sequential without sacrificing performance. This is less the case for 2 entire blocks.

Experimental results. Fig. 5 compares two approaches to parallelize computations: FFN and SA layers inside each block and parallelizing two entire blocks. The results demonstrate that both approaches perform well for QA tasks. However, parallelizing FFN and SA layers leads to significantly better results on all datasets.

Table 1: **Training with highly sparse LLMs**. We train the mapping module from scratch but with compressed LLM.

Method	Tr Time Reduction	#Param	#Tr Param	Sparsity	COCO ↑ CIDEr (test)	VQAv2 ↑ Acc (Val)
Baseline	×1 (31 min)	6.7B	7M	0.00	132.83	63.49
Wanda (task-specific)	×1	6.7B	7M	0.50	126.81	55.28
Random mask	×1	6.7B	7M	0.47	0.00	0.00
Magnitude (per-out)	×1	6.7B	7M	0.50	19.73	2.28
α-SubNet	×1	6.7B	7M	0.5	106.77	51.77
Random + Tr	–	6.7B	7M	0.70	111.43	33.58
Magnitude (per-out) + Tr	–	6.7B	7M	0.70	131.37	57.67
Wanda + Tr	–	6.7B	7M	0.50	131.15	64.49
Wanda + Tr	–	6.7B	7M	0.70	131.81	58.62
α-SubNet + Tr	–	6.7B	7M	0.70	132.07	56.75
Skip Block (T)	×1	6.7B	7M	0.5^*	66.26	61.56
Skip Block (P+T)	×1	6.7B	7M	0.5^*	20.21	0.13
Skip Block (P+T) + Tr	×1.5	6.7B	7M	0.5^*	131.53	59.13

4.3 Training with highly sparse LLMs

Training the mapping module from scratch as a remedy for skipping computations. As seen in previous sections, to preserve the original model performance, the amount of parameters to skip should be limited. In this context, we propose a remedy: training the mapping module from scratch with a compressed LLM. We maintain all hyperparameters and training details as the baseline models, with the only change being the LLM itself, where we either remove weights using Wanda [68] or α-SubNet [61], or skip entire blocks, applicable to all tokens.

Table 1 presents interesting results. Training with a compressed LLM achieves nearly the same performance for captioning and over 90% for VQAv2. This holds true when pruning 70% of the weights or skipping half the blocks (I=2). Moreover, in the case of the latter, besides enhancing inference efficiency, the training time is reduced by a factor of 1.5. For pruning methods, actual efficiency gains necessitate specialized hardware.

Comparison with other uncompressed models. To provide context for our findings, Table 2 offers a comparison with previous methods utilizing uncompressed LLMs. When training with a sparse LLM, more than 97% of the overall performance can be retained. Interestingly, despite having almost

Table 2: **Comparison between our compressed MLLMs and previous uncompressed ones.** Our models are competitive with previous SoTA despite skipping computations and having smaller number of trainable parameters.

Method	#P/#TP/Sparsity	Avg	COCO ↑	VQAv2 ↑	MSR-VTT ↑	MSVD-QA ↑	Audiocaps ↑
			CIDEr (test)	Acc (Val)	CIDEr (test)	Acc (test)	CIDEr (test)
ClipCAP [50]	7B/3.4M/0.00	–	113.08	–	–	–	–
MAPL [47]	7B/3.4M/0.00	–	125.2	43.5	–	–	–
eP-ALM [63]	6.7B/4M/0.00	63.11	111.6	54.9	48.79	38.40	61.86
DePALM [71]	7B/18.1M/0.00	–	131.29	70.11	49.88	–	69.70
Baseline	6.7B/7M/0.00	72.32	132.83	63.49	58.23	38.83	68.24
α-SubNet [61]	6.7B/7M/0.47	50.25 (69.48%)	106.77	51.77	38.37	31.19	23.15
α-SubNet + Tr	6.7B/7M/0.70	70.33 (97.24%)	132.07	56.75	57.56	38.77	66.52
Skip Block (P+T)	6.7B/7M/0.5*	6.28 (8.6%)	20.21	0.13	8.98	0.09	2.000
Skip Block (P+T) + Tr	6.7B/7M/0.5*	71.31 (98.6%)	131.53	59.13	58.31	39.08	68.52

half the trainable parameters and removing over 50% of model weights or blocks, our models are competitive with the current state-of-the-art approach (DePALM [71]) on nearly all tasks except VQAv2, where we underperform, and MSR-VTT, where we outperform. These results suggest that LLMs are significantly overparametrized for general multimodal tasks.

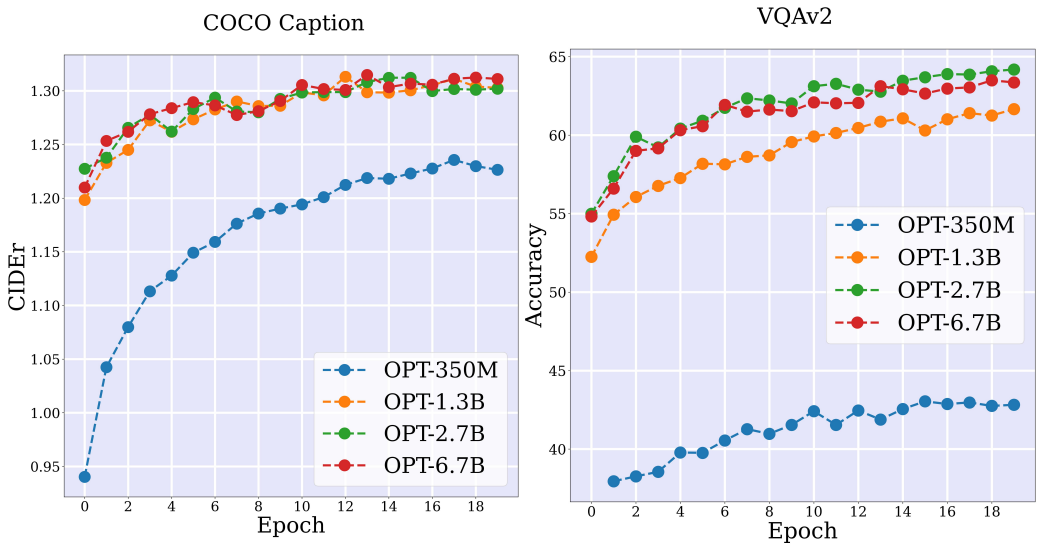

Figure 6: **Training with smaller LLMs.** Training with smaller versions of OPT models leads to comparable performance to larger ones.

4.4 Training with smaller LLMs

In previous sections, we explored the feasibility of significantly compressing LLMs for general multimodal tasks. Here, we delve into the possibility of training models with smaller LLMs.

Experimental results. Our models are trained with smaller versions of the OPT model family. We maintain the same training details as the baseline model, with the exception of the smallest OPT-350M model (where a smaller learning rate yields better results). Figure 6 presents a comparison between different LLM sizes. Interestingly, models up to 1.3B parameters exhibit the same performance for captioning. Similarly, for VQAv2, we can downscale the model to 2.7B parameters without sacrificing points. However, there is a notable gap, especially for VQAv2, between OPT-350M and other models. Additionally, compared to OPT-350M, larger models are more computationally efficient, as evidenced by the high scores after the first epoch. Table 3 further illustrates similar results across additional multimodal tasks. Particularly, the models with OPT-2.7B parameters compete with the baseline and previous approaches using larger LLMs. This suggests the feasibility of training with smaller LLMs, avoiding the high cost associated with larger ones. It is worth noting that previous works [63] have shown increasing performance with LLM size, but they trained with less powerful

visual encoders and for fewer epochs. We argue that proper training of the mapping module (e.g., better encoders, sufficient training) can diminish the improvement coming from larger LLMs.

Table 3: **Comparison with previous SoTA when training with smaller LLMs.** Training with OPT-2.7B leads to competitive performance compared to previous SoTA on diverse multimodal tasks.

Method	#P/#TP	Avg	COCO ↑	VQAv2 ↑	MSR-VTT ↑	MSVD-QA ↑	Audiocaps ↑
			CIDEr (test)	Acc (Val)	CIDEr (test)	Acc (test)	CIDEr (test)
ClipCAP [50]	7B/3.4M/0.00	–	113.08	–	–	–	–
MAPL [47]	7B/3.4M		125.2	43.5	–	–	–
eP-ALM [63]	6.7B/4M		111.6	54.9	48.79	38.40	61.86
DePALM [71]	7B/18.1M		131.29	70.11	49.88	–	69.70
Baseline (OPT-6.7B)	6.7B/7M		132.83	63.49	58.23	38.83	68.24
Baseline (OPT-2.7B)	2.7B/7M		132.83	64.18	57.33	40.29	65.47

4.5 Case study for larger scale multitask MLLMs.

In this section, we investigate the generalization of the proposed approaches to the larger scale multitask setup. We focus on the LLaVA-1.5 [39] model which consists of CLIP-ViT-L, Vicuna-v1.5 connected with an MLP and trained on a collection of public datasets. This model is evaluated on more recent multimodal benchmarks such as SEED [33], MME [21], POPE [36] ScienceQA (SQA) and [44], but also on traditional benchmarks such as VQAv2 [24], GQA [27] and TextVQA [66].

Experimental results. Table 4 shows that we can remove 50% of the parameters and maintain more than 90% of the original performance across several datasets. Computing the pruning score based on all tokens gives the best results, followed by the prompt (the contains the visual tokens) and then the textual tokens. We also propose to skip entire blocks (Fig. 7). Similar to previous section, skipping only the generated textual tokens leads to the best results where we can retain more than 90% of the original performance. In general, we notice larger degradation in performance compared to the single task setup (Sec. 4). However that results also suggests the possibility of removing redundant computations for large scale models.

Table 4: **Skipping computations for LLaVA-1.5.** Left: Post-training pruning with Wanda by keeping the weights that are mostly activated by the: prompt (P), textual (T) or all (P+T) tokens.

Method	Sparsity	GQA ↑	VQAv2 ↑	TextVQA ↑	SQA-IMG ↑	COCO ↑	POPE ↑
		Acc (test-dev)	Acc (test-dev)	Acc (Val)	Acc (test)	CIDEr (test)	Acc (test)
LLaVA-1.5	0.0 %	61.96	78.50	58.20	66.8	110.49	85.96
LLaVA-1.5-Wanda (P + T)	50 %	51.15 (82.58 %)	71.31 (90.82 %)	46.64 (80.06 %)	60.24 (90.23 %)	100.04 (90.52 %)	87.22 (–%)
LLaVA-1.5-Wanda (P)	50 %	48.1 (77.57 %)	68.76 (87.65 %)	41.94 (72.06 %)	54.44 (81.58 %)	76.99 (69.58 %)	85.37 (99.20 %)
LLaVA-1.5-Wanda (T)	50 %	43.93 (70.92 %)	64.11 (81.75 %)	41.74 (71.70 %)	51.86 (77.78 %)	94.09 (85.10 %)	78.28 (90.97 %)

5 Discussion

Single-task vs multi-task MLLMs. In this work, we focus on parameter and data-efficient models in a single-task setup showing high amount of computation redundancy. We also complement our work with preliminary results on larger-scale models such as LLaVA-1.5 that are trained in multitask fashion and can support wider range of tasks, including conducting dialog with humans. On this setup, we also show similar observations, however we think these baselines requires more adapted compression techniques to limit the performance degradation.

Dynamic compute. In our study, we focus on static computation skipping techniques, where the skipping strategy remains constant regardless of the task or input example. These static approaches are hardware-agnostic and compatible with scaling techniques. However, an alternative direction involves exploring more adaptive compute strategies, which ideally allocate varying amounts of computations based on the task's complexity. While similar approaches have been proposed for LLMs [1, 57, 32, 4, 59, 55], we believe there is still significant room for improvement in this area. Ultimately, we view our study as an initial step in highlighting the overparameterization of LLMs and advocating for greater efforts to reduce their computational costs.

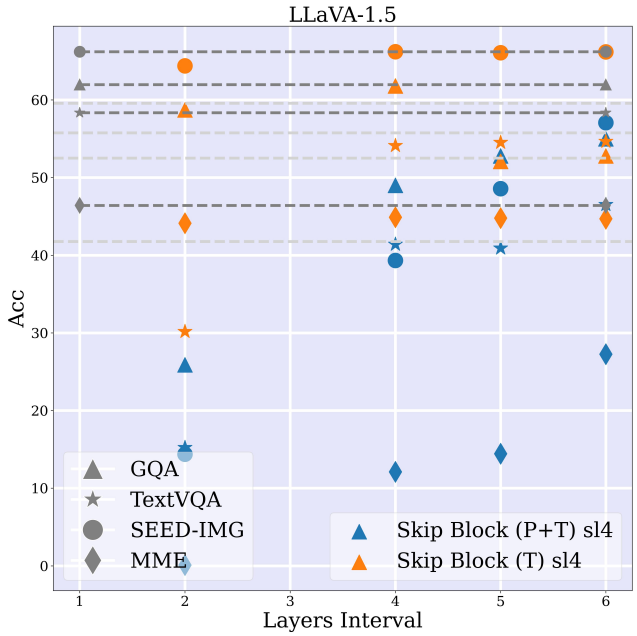

Figure 7: **Skipping computations for LLaVA-1.5.** Skipping entire blocks for textual (T) and all tokens (P+T), each interval of layers. The gray and light gray indicate the original performance and 90% of it.

Limitations. Our study primarily focuses on parameter-efficient MLLMs, where the LLM remains frozen. We acknowledge that there are other architectures we did not explore, such as those involving interaction through cross-attention mechanisms [2, 31]. Besides, we did not delve into more complex multimodal tasks that necessitate reasoning capabilities [79]. Because the main objective of the paper does not include proposing new SoTA model compression technique, we did not extensively compare with more recent approaches. Extending our study to encompass these scenarios represents an important avenue for future research. In addition, we focus on investigating the redundancy that could lead theoretically to high efficient training and inference. However, the reduction is expected to be smaller at actual devices as there are many other affecting factors.

6 Conclusion

This study investigates the redundancy of computations in perceptually augmented LLMs (MLLMs) across various granularity levels. Our experiments reveal the potential for significant reduction in computations by skipping entire blocks, FFN layers, and even individual neurons. We demonstrate that training the mapping module with severely compressed LLMs can effectively preserve over 97% of performance. Alternatively, training with smaller LLMs can achieve comparable performance to models two or three times larger. We show similar findings across both single-task and multitask multimodal settings, underscoring their broad applicability. We hope that this work will encourage future works to focus on methods to reduce the computation cost of MLLMs at both training and inference stages.

7 Acknowledgments

The authors would like to thank Arnaud Dapogny and Edouard Yvinec for fruitful discussions, and Damien Teney and Alexandre Ramé for their helpful feedback on the paper. This work was partly supported by ANR grant VISA DEEP (ANR-20-CHIA-0022), and HPC resources of IDRIS under the allocation 2024-[AD011013415R2] made by GENCI.

References

[1] Ainslie, J., Lei, T., de Jong, M., Ontañón, S., Brahma, S., Zemlyanskiy, Y., Uthus, D., Guo, M., Lee-Thorp, J., Tay, Y., et al.: Colt5: Faster long-range transformers with conditional computation. arXiv preprint arXiv:2303.09752 (2023) 3, 8

[2] Alayrac, J.B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al.: Flamingo: a visual language model for few-shot learning. Advances in Neural Information Processing Systems (NeurIPS) **35**, 23716–23736 (2022) 1, 2, 9

[3] Awadalla, A., Gao, I., Gardner, J., Hessel, J., Hanafy, Y., Zhu, W., Marathe, K., Bitton, Y., Gadre, S., Sagawa, S., et al.: OpenFlamingo: An open-source framework for training large autoregressive vision-language models. arXiv preprint arXiv:2308.01390 (2023) 2

[4] Bengio, E., Bacon, P.L., Pineau, J., Precup, D.: Conditional computation in neural networks for faster models. arXiv preprint arXiv:1511.06297 (2015) 8

[5] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in Neural Information Processing Systems (NeurIPS) **33**, 1877–1901 (2020) 2, 3

[6] Chee, J., Cai, Y., Kuleshov, V., De Sa, C.M.: Quip: 2-bit quantization of large language models with guarantees. Advances in Neural Information Processing Systems **36** (2024) 3

[7] Chen, X., Djolonga, J., Padlewski, P., Mustafa, B., Changpinyo, S., Wu, J., Ruiz, C.R., Goodman, S., Wang, X., Tay, Y., et al.: Pali-x: On scaling up a multilingual vision and language model. arXiv preprint arXiv:2305.18565 (2023) 1, 2

[8] Chen, X., Wang, X., Changpinyo, S., Piergiovanni, A., Padlewski, P., Salz, D., Goodman, S., Grycner, A., Mustafa, B., Beyer, L., et al.: PaLI: A jointly-scaled multilingual language-image model. arXiv preprint arXiv:2209.06794 (2022) 2

[9] Chen, Y., Pan, X., Li, Y., Ding, B., Zhou, J.: Ee-llm: Large-scale training and inference of early-exit large language models with 3d parallelism. arXiv preprint arXiv:2312.04916 (2023) 3

[10] Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H.W., Sutton, C., Gehrmann, S., et al.: Palm: Scaling language modeling with pathways. arXiv preprint arXiv:2204.02311 (2022) 1, 2

[11] Dai, W., Li, J., Li, D., Tiong, A.M.H., Zhao, J., Wang, W., Li, B., Fung, P., Hoi, S.: Instructblip: Towards general-purpose vision-language models with instruction tuning. arXiv preprint arXiv:2305.06500 (2023) 1, 2

[12] Diao, S., Zhou, W., Zhang, X., Wang, J.: Write and paint: Generative vision-language models are unified modal learners. In: The Eleventh International Conference on Learning Representations (2023), https://openreview.net/forum?id=HgQR0mXQ1_a 2

[13] Dou, Z.Y., Xu, Y., Gan, Z., Wang, J., Wang, S., Wang, L., Zhu, C., Liu, Z., Zeng, M., et al.: An empirical study of training end-to-end vision-and-language transformers. arXiv preprint arXiv:2111.02387 (2021) 2

[14] Driess, D., Xia, F., Sajjadi, M.S., Lynch, C., Chowdhery, A., Ichter, B., Wahid, A., Tompson, J., Vuong, Q., Yu, T., et al.: Palm-e: An embodied multimodal language model. arXiv preprint arXiv:2303.03378 (2023) 1, 2

[15] Egiazarian, V., Panferov, A., Kuznedelev, D., Frantar, E., Babenko, A., Alistarh, D.: Extreme compression of large language models via additive quantization. arXiv preprint arXiv:2401.06118 (2024) 3

[16] Fang, Z., Wang, J., Hu, X., Wang, L., Yang, Y., Liu, Z.: Compressing visual-linguistic model via knowledge distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1428–1438 (2021) 3

[17] Frantar, E., Alistarh, D.: Optimal brain compression: A framework for accurate post-training quantization and pruning. Advances in Neural Information Processing Systems **35**, 4475–4488 (2022) 3

[18] Frantar, E., Alistarh, D.: Spdy: Accurate pruning with speedup guarantees. In: International Conference on Machine Learning. pp. 6726–6743. PMLR (2022) 3

[19] Frantar, E., Alistarh, D.: Sparsegpt: Massive language models can be accurately pruned in one-shot. In: International Conference on Machine Learning. pp. 10323–10337. PMLR (2023) 3

[20] Frantar, E., Ashkboos, S., Hoefler, T., Alistarh, D.: OPTQ: Accurate quantization for generative pre-trained transformers. In: The Eleventh International Conference on Learning Representations (2023), https://openreview.net/forum?id=tcbBPnfwxS 3

[21] Fu, C., Chen, P., Shen, Y., Qin, Y., Zhang, M., Lin, X., Qiu, Z., Lin, W., Yang, J., Zheng, X., et al.: Mme: A comprehensive evaluation benchmark for multimodal large language models. arXiv preprint arXiv:2306.13394 (2023) 8

[22] Gan, Z., Chen, Y.C., Li, L., Chen, T., Cheng, Y., Wang, S., Liu, J., Wang, L., Liu, Z.: Playing lottery tickets with vision and language. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 652–660 (2022) 3

[23] Gong, Y., Chung, Y.A., Glass, J.: AST: Audio Spectrogram Transformer. In: Proc. Interspeech 2021. pp. 571–575 (2021). https://doi.org/10.21437/Interspeech.2021-698 4

[24] Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., Parikh, D.: Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6904–6913 (2017) 4, 8

[25] Gromov, A., Tirumala, K., Shapourian, H., Glorioso, P., Roberts, D.A.: The unreasonable ineffectiveness of the deeper layers. arXiv preprint arXiv:2403.17887 (2024) 2, 4

[26] Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., Casas, D.d.L., Hendricks, L.A., Welbl, J., Clark, A., et al.: Training compute-optimal large language models. arXiv preprint arXiv:2203.15556 (2022) 1, 2

[27] Hudson, D.A., Manning, C.D.: Gqa: A new dataset for real-world visual reasoning and compositional question answering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6700–6709 (2019) 8

[28] Kim, C.D., Kim, B., Lee, H., Kim, G.: Audiocaps: Generating captions for audios in the wild. In: NAACL-HLT (2019) 4

[29] Koh, J.Y., Salakhutdinov, R., Fried, D.: Grounding language models to images for multimodal generation. arXiv preprint arXiv:2301.13823 (2023) 2

[30] Kwon, W., Kim, S., Mahoney, M.W., Hassoun, J., Keutzer, K., Gholami, A.: A fast post-training pruning framework for transformers. In: Oh, A.H., Agarwal, A., Belgrave, D., Cho, K. (eds.) Advances in Neural Information Processing Systems (2022), https://openreview.net/forum?id=0GRBKLBjJE 3

[31] Laurençon, H., Saulnier, L., Tronchon, L., Bekman, S., Singh, A., Lozhkov, A., Wang, T., Karamcheti, S., Rush, A.M., Kiela, D., Cord, M., Sanh, V.: OBELICS: An open web-scale filtered dataset of interleaved image-text documents. In: Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track (2023), https://openreview.net/forum?id=SKN2hflBIZ 2, 9

[32] Lei, T., Bai, J., Brahma, S., Ainslie, J., Lee, K., Zhou, Y., Du, N., Zhao, V., Wu, Y., Li, B., et al.: Conditional adapters: Parameter-efficient transfer learning with fast inference. Advances in Neural Information Processing Systems **36** (2024) 8

[33] Li, B., Wang, R., Wang, G., Ge, Y., Ge, Y., Shan, Y.: Seed-bench: Benchmarking multimodal llms with generative comprehension. arXiv preprint arXiv:2307.16125 (2023) 8

[34] Li, J., Li, D., Xiong, C., Hoi, S.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. arXiv preprint arXiv:2201.12086 (2022) 2

[35] Li, J., Selvaraju, R., Gotmare, A., Joty, S., Xiong, C., Hoi, S.C.H.: Align before fuse: Vision and language representation learning with momentum distillation. Advances in Neural Information Processing Systems (NeurIPS) **34** (2021) 2

[36] Li, Y., Du, Y., Zhou, K., Wang, J., Zhao, W.X., Wen, J.R.: Evaluating object hallucination in large vision-language models. arXiv preprint arXiv:2305.10355 (2023) 8

[37] Liang, S., Zhao, M., Schütze, H.: Modular and parameter-efficient multimodal fusion with prompting. In: Findings of the Association for Computational Linguistics: ACL 2022. pp. 2976–2985 (2022) 3

[38] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European Conference on Computer Vision (ECCV). pp. 740–755. Springer (2014) 4

[39] Liu, H., Li, C., Li, Y., Lee, Y.J.: Improved baselines with visual instruction tuning. arXiv preprint arXiv:2310.03744 (2023) 8

[40] Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. arXiv preprint arXiv:2304.08485 (2023) 1

[41] Liu, Z., Wang, J., Dao, T., Zhou, T., Yuan, B., Song, Z., Shrivastava, A., Zhang, C., Tian, Y., Re, C., et al.: Deja vu: Contextual sparsity for efficient llms at inference time. In: International Conference on Machine Learning. pp. 22137–22176. PMLR (2023) 2, 3, 4

[42] Lu, J., Clark, C., Lee, S., Zhang, Z., Khosla, S., Marten, R., Hoiem, D., Kembhavi, A.: Unified-io 2: Scaling autoregressive multimodal models with vision, language, audio, and action. arXiv preprint arXiv:2312.17172 (2023) 2

[43] Lu, J., Clark, C., Zellers, R., Mottaghi, R., Kembhavi, A.: Unified-io: A unified model for vision, language, and multi-modal tasks. In: The Eleventh International Conference on Learning Representations (2022) 2

[44] Lu, P., Mishra, S., Xia, T., Qiu, L., Chang, K.W., Zhu, S.C., Tafjord, O., Clark, P., Kalyan, A.: Learn to explain: Multimodal reasoning via thought chains for science question answering. Advances in Neural Information Processing Systems **35**, 2507–2521 (2022) 8

[45] Ma, X., Fang, G., Wang, X.: Llm-pruner: On the structural pruning of large language models. Advances in neural information processing systems **36**, 21702–21720 (2023) 3

[46] Ma, Y., Xu, G., Sun, X., Yan, M., Zhang, J., Ji, R.: X-clip: End-to-end multi-grained contrastive learning for video-text retrieval. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 638–647 (2022) 4

[47] Mañas, O., Rodriguez, P., Ahmadi, S., Nematzadeh, A., Goyal, Y., Agrawal, A.: Mapl: Parameter-efficient adaptation of unimodal pre-trained models for vision-language few-shot prompting. arXiv preprint arXiv:2210.07179 (2022) 1, 2, 3, 7, 8

[48] Merullo, J., Castricato, L., Eickhoff, C., Pavlick, E.: Linearly mapping from image to text space. arXiv preprint arXiv:2209.15162 (2022) 1, 2

[49] Mizrahi, D., Bachmann, R., Kar, O.F., Yeo, T., Gao, M., Dehghan, A., Zamir, A.: 4m: Massively multimodal masked modeling. In: Thirty-seventh Conference on Neural Information Processing Systems (2023) 2

[50] Mokady, R., Hertz, A., Bermano, A.H.: Clipcap: Clip prefix for image captioning. arXiv preprint arXiv:2111.09734 (2021) 7, 8

[51] Moon, S., Madotto, A., Lin, Z., Nagarajan, T., Smith, M., Jain, S., Yeh, C.F., Murugesan, P., Heidari, P., Liu, Y., et al.: Anymal: An efficient and scalable any-modality augmented language model. arXiv preprint arXiv:2309.16058 (2023) 2

[52] OpenAI: Gpt-4 technical report. arXiv (2023) 1, 2

[53] Panagopoulou, A., Xue, L., Yu, N., Li, J., Li, D., Joty, S., Xu, R., Savarese, S., Xiong, C., Niebles, J.C.: X-instructblip: A framework for aligning x-modal instruction-aware representations to llms and emergent cross-modal reasoning. arXiv preprint arXiv:2311.18799 (2023) 3

[54] Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning (ICML). pp. 8748–8763. PMLR (2021) 4

[55] Raposo, D., Ritter, S., Richards, B., Lillicrap, T., Humphreys, P.C., Santoro, A.: Mixture-of-depths: Dynamically allocating compute in transformer-based language models. arXiv preprint arXiv:2404.02258 (2024) 3, 8

[56] Scao, T.L., Fan, A., Akiki, C., Pavlick, E., Ilić, S., Hesslow, D., Castagné, R., Luccioni, A.S., Yvon, F., Gallé, M., et al.: Bloom: A 176b-parameter open-access multilingual language model. arXiv preprint arXiv:2211.05100 (2022) 2, 3

[57] Schuster, T., Fisch, A., Gupta, J., Dehghani, M., Bahri, D., Tran, V., Tay, Y., Metzler, D.: Confident adaptive language modeling. Advances in Neural Information Processing Systems **35**, 17456–17472 (2022) 3, 8

[58] Shao, W., Chen, M., Zhang, Z., Xu, P., Zhao, L., Li, Z., Zhang, K., Gao, P., Qiao, Y., Luo, P.: Omniquant: Omnidirectionally calibrated quantization for large language models. arXiv preprint arXiv:2308.13137 (2023) 3

[59] Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., Dean, J.: Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In: International Conference on Learning Representations (2016) 3, 8

[60] Shi, D., Tao, C., Jin, Y., Yang, Z., Yuan, C., Wang, J.: UPop: Unified and progressive pruning for compressing vision-language transformers. In: Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., Scarlett, J. (eds.) Proceedings of the 40th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 202, pp. 31292–31311. PMLR (23–29 Jul 2023), https://proceedings.mlr.press/v202/shi23e.html 3

[61] Shukor, M., Cord, M.: Implicit multimodal alignment: On the generalization of frozen llms to multimodal inputs. arXiv preprint arXiv:2405.16700 (2024) 2, 3, 4, 5, 6, 7

[62] Shukor, M., Couairon, G., Cord, M.: Efficient vision-language pretraining with visual concepts and hierarchical alignment. In: 33rd British Machine Vision Conference (BMVC) (2022) 2

[63] Shukor, M., Dancette, C., Cord, M.: ep-alm: Efficient perceptual augmentation of language models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 22056–22069 (October 2023) 1, 2, 3, 7, 8

[64] Shukor, M., Dancette, C., Rame, A., Cord, M.: UnIVAL: Unified model for image, video, audio and language tasks. Transactions on Machine Learning Research (2023), https://openreview.net/forum?id=4uflhObpcp 2

[65] Singh, A., Hu, R., Goswami, V., Couairon, G., Galuba, W., Rohrbach, M., Kiela, D.: Flava: A foundational language and vision alignment model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 15638–15650 (2022) 2

[66] Singh, A., Natarajan, V., Shah, M., Jiang, Y., Chen, X., Batra, D., Parikh, D., Rohrbach, M.: Towards vqa models that can read. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8317–8326 (2019) 8

[67] Song, J., Oh, K., Kim, T., Kim, H., Kim, Y., Kim, J.J.: Sleb: Streamlining llms through redundancy verification and elimination of transformer blocks. arXiv preprint arXiv:2402.09025 (2024) 2, 4

[68] Sun, M., Liu, Z., Bair, A., Kolter, J.Z.: A simple and effective pruning approach for large language models. arXiv preprint arXiv:2306.11695 (2023) 3, 6

[69] Tan, J.H., Chan, C.S., Chuah, J.H.: End-to-end supermask pruning: Learning to prune image captioning models. Pattern Recognition **122**, 108366 (2022) 3

[70] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al.: Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288 (2023) 1, 2

[71] Vallaeys, T., Shukor, M., Cord, M., Verbeek, J.: Improved baselines for data-efficient perceptual augmentation of llms. arXiv preprint arXiv:2403.13499 (2024) 1, 3, 4, 7, 8

[72] Wang, P., Yang, A., Men, R., Lin, J., Bai, S., Li, Z., Ma, J., Zhou, C., Zhou, J., Yang, H.: Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In: International Conference on Machine Learning. pp. 23318–23340. PMLR (2022) 2

[73] Wang, Z., Wang, L., Zhao, Z., Wu, M., Lyu, C., Li, H., Cai, D., Zhou, L., Shi, S., Tu, Z.: Gpt4video: A unified multimodal large language model for lnstruction-followed understanding and safety-aware generation. arXiv preprint arXiv:2311.16511 (2023) 3

[74] Wu, K., Peng, H., Zhou, Z., Xiao, B., Liu, M., Yuan, L., Xuan, H., Valenzuela, M., Chen, X.S., Wang, X., et al.: Tinyclip: Clip distillation via affinity mimicking and weight inheritance. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 21970–21980 (2023) 3

[75] Wu, S., Fei, H., Qu, L., Ji, W., Chua, T.S.: Next-gpt: Any-to-any multimodal llm. arXiv preprint arXiv:2309.05519 (2023) 3

[76] Xu, D., Zhao, Z., Xiao, J., Wu, F., Zhang, H., He, X., Zhuang, Y.: Video question answering via gradually refined attention over appearance and motion. In: Proceedings of the 25th ACM International Conference on Multimedia. p. 1645–1653. MM '17, Association for Computing Machinery, New York, NY, USA (2017). https://doi.org/10.1145/3123266.3123427, https://doi.org/10.1145/3123266.3123427 4

[77] Xu, J., Mei, T., Yao, T., Rui, Y.: Msr-vtt: A large video description dataset for bridging video and language. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016) 4

[78] Yang, A., Miech, A., Sivic, J., Laptev, I., Schmid, C.: Zero-shot video question answering via frozen bidirectional language models. In: NeurIPS 2022-36th Conference on Neural Information Processing Systems (2022) 1, 2

[79] Yue, X., Ni, Y., Zhang, K., Zheng, T., Liu, R., Zhang, G., Stevens, S., Jiang, D., Ren, W., Sun, Y., et al.: Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. arXiv preprint arXiv:2311.16502 (2023) 9

[80] Yvinec, E., Dapogny, A., Cord, M., Bailly, K.: Rex: Data-free residual quantization error expansion. Advances in Neural Information Processing Systems **36** (2024) 3

[81] Zhang, R., Han, J., Zhou, A., Hu, X., Yan, S., Lu, P., Li, H., Gao, P., Qiao, Y.J.: LLaMA-Adapter: Efficient fine-tuning of language models with zero-init attention **2303.16199** (2023), https://api.semanticscholar.org/CorpusID:257771811 2

[82] Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X.V., et al.: Opt: Open pre-trained transformer language models. arXiv preprint arXiv:2205.01068 (2022) 1, 2, 3, 4

[83] Zheng, L., Chiang, W.L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., et al.: Judging llm-as-a-judge with mt-bench and chatbot arena. Advances in Neural Information Processing Systems **36** (2024) 4

[84] Zhu, D., Chen, J., Shen, X., Li, X., Elhoseiny, M.: Minigpt-4: Enhancing vision-language understanding with advanced large language models. arXiv preprint arXiv:2304.10592 (2023) 2