Beyond CLIP: Adapting Stable Diffusion for Multimodal Tasks with LLM

Shuheng Zhang, Weihao Ye, Yuhang Hu

class AI

36920231153262@stu.xmu.edu.cn, 36920231153257@stu.xmu.edu.cn, 36920231153194@stu.xmu.edu.cn

Abstract

Recently, large language models (LLMs) have garnered significant attention due to their advancements in natural language processing, demonstrating unparalleled capabilities in text understanding and generation. Furthermore, research on synthesizing high-fidelity complex images from text has been extremely popular in the past two years. However, research on multimodal large language models (MM-LLMs) is just beginning. If existing well-performing encoders or decoders can be fully utilized in MM-LLMs, it would significantly reduce the training cost. This paper illustrates how we can integrate a large language model with a diffusion model for Text-to-Image (T2I) by fine-tuning less than 1% of parameters. This integration enables image generation within a conversation, achieving a quality comparable to the original diffusion model. Simultaneously, through a similar fine-tuning approach, more modalities (such as audio and video) can be aligned with the large language model to obtain a feature-rich multimodal large language model.

Introduction

In the recent past, certain Artificial Intelligence Generated Content (AIGC) technologies have witnessed unparalleled advancements, including ChatGPT for textual generation(OpenAI 2023) and diffusion models(Fan et al. 2023) for visual creation. Notably, the emergence of large language models (LLMs) such as Flan-T5(Chung et al. 2022), Vicuna(Chiang et al. 2023), Llama(Touvron et al. 2023), and Alpaca(Taori et al. 2023) has been remarkable. These models exhibit their impressive capabilities of language reasoning and decision-making at a human level, illuminating the route towards Artificial General Intelligence (AGI). Fundamentally, our world is multimodal. Humans interact with the world using various sensory organs that perceive different modalities of information (like language, images, videos, sounds, etc.), and these information modalities often supplement and work in tandem with each other. Guided by this understanding, LLMs that were initially text-based have recently been equipped with the ability to understand and perceive other modalities such as vision, video, audio, etc.

An approach worth mentioning is the use of adapters to synchronize other types of pre-trained encoders with text

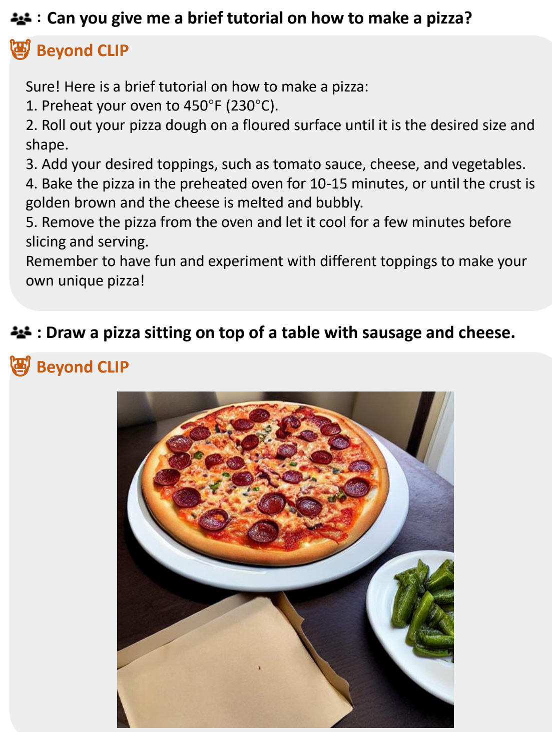

Figure 1: Our model is not only capable of engaging in normal conversations but with minor modifications, it can also generate images.

language models. This initiative has spurred the swift evolution of multimodal LLMs (MM-LLMs), including models like BLIP-2(Li et al. 2023a), Flamingo(Alayrac et al. 2022), MiniGPT-4(Zhu et al. 2023), Video-LLaMA(Zhang, Li, and Bing 2023), LLaVA(Liu et al. 2023), PandaGPT(Su et al. 2023), SpeechGPT(Zhang et al. 2023b). However, these studies primarily concentrate on understanding multimodal content at the input stage and cannot produce content in multiple modalities beyond text. We underscore that genuine human cognition and communication indispensably necessitate fluid transitions between any information modalities. This necessitates the investigation of any-to-any MM-LLM as a vital step towards realizing true AGI, which involves accepting input in any modality and delivering responses in a suitable form of any modality.

Initiatives have been undertaken to emulate the any-to-any mode transformation akin to human beings. Recently, advancements have been made by CoDi(Tang et al. 2023) in the capability to process and generate any mix of modalities concurrently. However, it falls short in terms of the reasoning and decision-making capabilities inherent in the LLM and is confined to basic pairwise content creation. Conversely, attempts such as Visual-ChatGPT(Wu et al. 2023) and HuggingGPT(Shen et al. 2023) strive to amalgamate LLM with external instruments to accomplish an approximate 'any-to-any' multimodal comprehension and generation. Regrettably, owing to the comprehensive pipeline architecture, these systems encounter significant obstacles. Primarily, the information exchange between different modules relies entirely on the discrete text produced by LLM, where the cascading procedure inevitably leads to noise and propagation errors. More crucially, the entire system solely employs existing pre-trained tools for inference. Owing to the absence of comprehensive end-to-end training in error propagation, the capacity for content comprehension and multimodal generation might be considerably restricted. In essence, there is a pressing requirement to construct an end-to-end MM-LLM for any modality.

To achieve such a goal, we first attempt the T2I task. Initially, we use an existing open-source LLM(Touvron et al. 2023) as the core for semantic understanding and reasoning. This LLM not only generates text tokens but also determines whether to convert these tokens into text or pass them into the mapping layer to enter the subsequent diffusion model(Rombach et al. 2022), as needed. As shown in Figure 1.

Training such a model from scratch requires an extremely large amount of computational resources and does not fully utilize the existing pre-trained weights. Therefore, in the method of this paper, we make full use of the well-performing pre-trained LLM and diffusion model. By loading ready-made parameters, we not only avoid cold start training but also provide valuable experience for subsequent work on more modalities. To align the features of the LLM and diffusion model, we only train some adapter layers and mapping layers, the parameter volume of which is less than 1% of the model parameters, and all other pre-trained parameters are frozen. We found that through such operations, comparable results to the original diffusion model can be achieved on the T2I task. In summary, the contributions of this paper are as follows:

- We propose a novel approach to integrate LLMs with diffusion models as a component of MM-LLMs. Based on this approach, we can also combine LLMs with other modalities similarly for subsequent tasks.
- We employ a lightweight alignment learning technique that only trains some adapter layers and mapping layers and requires minimal parameter adjustments (only 1% of the parameters) to achieve effective semantic alignment.

Related Work

Text-to-Image Large Models

Lately, substantial pre-trained autoregressive and diffusion models have exhibited exceptional outcomes in the realm of text-to-image synthesis. Models like DALL-E(Ramesh et al. 2021), CogView(Ding et al. 2021), and M6(Lin et al. 2021) employ VQ-VAE(Van Den Oord, Vinyals et al. 2017) or VQ-GAN(Esser, Rombach, and Ommer 2021) to transcribe images into distinct image tokens. These tokens, in conjunction with word tokens, are subsequently utilized to pre-train a significant unidirectional transformer for autoregressive generation. Parti(Yu et al. 2022) presents a sequence-to-sequence autoregressive model to perceive text-to-image synthesis as a translation task. Cogview2(Ding et al. 2022) uses hierarchical transformers and local parallel autoregressive generation for accelerated autoregressive image generation. Certain studies endeavor to apply the diffusion model(Dhariwal and Nichol 2021; Ho, Jain, and Abbeel 2020; Ho et al. 2022; Nichol and Dhariwal 2021; Sohl-Dickstein et al. 2015) to rectify the slow generation drawback of the autoregressive model. VQ-Diffusion(Gu et al. 2022) amalgamates the VQ-VAE(Van Den Oord, Vinyals et al. 2017) and diffusion model(Ho et al. 2022; Nichol and Dhariwal 2021) to eliminate the unidirectional bias and avert accumulated prediction errors. GLIDE(Nichol et al. 2021) employs guided diffusion to address the issue of text-conditional image synthesis. DALL-E2(Ramesh et al. 2022) combines the CLIP representation and diffusion model to construct a CLIP decoder. Latent Diffusion Models (LDM)(Rombach et al. 2022) implement the diffusion model in the latent space to enable training on restricted computational resources while preserving image quality. A particular text-to-image LDM is Stable Diffusion, a well-liked open-source project that provides an easy-to-use interface. Imagen(Saharia et al. 2022) integrates the large language model(Raffel et al. 2020) to produce high-quality text features and proposes an Efficient U-Net for diffusion models.

Multimodal Large Language Models

LLMs have already catalyzed substantial shifts and innovations throughout the AI community and beyond. The most prominent LLMs, specifically OpenAI's ChatGPT and GPT4(OpenAI 2023), armed with alignment techniques such as instruction tuning(Ouyang et al. 2022; Li et al. 2023c; Zhang et al. 2023c; Liu et al. 2023) and reinforcement learning from human feedback (RLHF)(Stiennon et al. 2020), have exhibited extraordinary competencies in language understanding and reasoning. Moreover, a range of open-source LLMs, such as Flan-T5(Chung et al. 2022), Vicuna(Chiang et al. 2023), LLaMA(Touvron et al. 2023), and Alpaca(Taori et al. 2023), have notably accelerated advancements and made significant contributions to the community(Zhu et al. 2023; Zhang et al. 2023a). Following this, substantial efforts have been invested in engineering LLMs adept at managing multimodal inputs and tasks, leading to the advent of MM-LLMs.

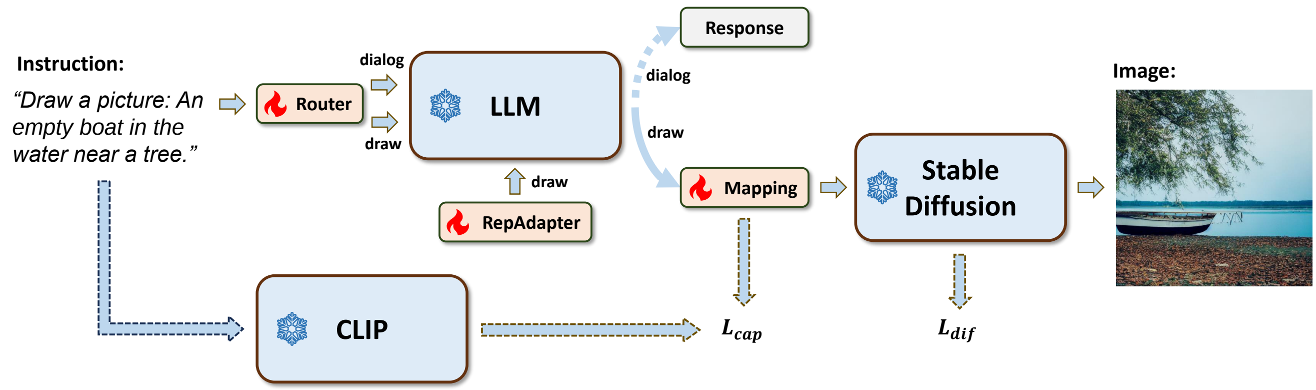

Figure 2: The model can generate either text or images, depending on the specific requirements. If only normal conversations using the LLM are needed, then only this frozen LLM is used. If the model believes that an image needs to be generated, then the fine-tuned LLM is used as the text encoder for stable diffusion. Throughout the training process, less than 1% of the parameters need to be adjusted, and its generation effect is comparable to the original version of stable diffusion.

On one hand, a large number of researchers are developing fundamental MM-LLMs by aligning the efficiently trained encoders of various modalities to the textual feature space of LLMs, thus allowing LLMs to interpret inputs from other modalities(Huang et al. 2023; Zhu et al. 2023; Su et al. 2022; Koh, Fried, and Salakhutdinov 2023). For example, Flamingo(Alayrac et al. 2022) employs a cross-attention layer to associate a fixed image encoder with the LLMs. BLIP-2(Li et al. 2023a) leverages a Q-Former to transform the input image queries into the LLMs. LLaVA(Liu et al. 2023) uses a simple projection scheme to link image features to the word embedding space. There are also several similar methods for creating MM-LLMs that can understand videos (e.g., Video-Chat(Li et al. 2023b) and Video-LLaMA(Zhang, Li, and Bing 2023)), audio (e.g., SpeechGPT(Zhang et al. 2023b)), etc. Importantly, PandaGPT(Su et al. 2023) achieves a comprehensive understanding of six different modalities at once by integrating the multimodal encoder, i.e., ImageBind(Girdhar et al. 2023).

However, these MM-LLMs are all constrained by the limitation of only perceiving multimodal data and not generating content in arbitrary modalities. In this paper, we concentrate on the aspect of content generation. Using image generation as an example, we explore the impact of fine-tuning a minimal number of parameters on multimodal generation tasks.

Method

This section primarily consists of two parts. Firstly, we will provide a brief introduction to the LLaMA(Touvron et al. 2023), RepAdapter(Luo et al. 2023), and stable diffusion(Rombach et al. 2022), which will be utilized in the subsequent sections. Following that, we will delve into a detailed explanation of the overall structure of the model proposed in this paper.

Background Knowledge

LLaMA. Meta AI's LLaMA 2(Touvron et al. 2023) is an open-source large language model for research and commercial use. It includes pre-trained and fine-tuned models (LLaMA Chat, Code LLaMA) with parameters from 7 to 70 billion. The pre-trained model was trained on 2 trillion tokens, double the context length of LLaMA 1, and fine-tuned models were trained on over 1 million human annotations.

LLaMA 2 surpasses other open-source models in various benchmarks, including inference, coding, proficiency, and knowledge tests. LLaMA Chat is pre-trained on public online data, and fine-tuned models use public instruction datasets and over 1 million human annotations. Code LLaMA, a code generation model, is built on LLaMA 2 and trained on 500 billion code tokens.

RepAdapter. RepAdapter(Luo et al. 2023) is a parameter-efficient adapter for large visual models, integrating into most visual models through structural reparameterization. It achieves zero cost in the inference process and gains advantages in parameter efficiency and performance through sparse design and effective placement of the adapter structure.

RepAdapter's performance and efficiency were validated on 27 benchmark datasets of three visual tasks. It outperforms full adjustment by 7.2% on average, saving up to 25% of training time, 20% of GPU memory, and 94.6% of storage cost of ViT-B/16 on VTAB-1k. Its generalization ability has been validated by a series of visual models.

Stable Diffusion. Stable Diffusion, a Latent Diffusion Model (LDM) implementation, is a high-resolution image synthesis model. It generates images by iteratively denoising data in latent space and decoding the results into an image. This allows image generation on consumer-grade GPUs in seconds, lowering deployment thresholds.

These iterative models take random noise, conditioned by text or images, as input. They learn to remove this noise iteratively, resulting in a final image.

These models can be trained on limited resources while maintaining quality and flexibility. The introduction of cross-attention layers transforms the diffusion model into a powerful generator for general conditional inputs like text or bounding boxes, and high-resolution synthesis. LDMs have achieved a new technical level in image restoration and competitive performance in tasks like unconditional image generation, semantic scene synthesis, and super-resolution, with significantly reduced computational requirements compared to pixel-based DMs.

Model Design

As shown in Figure 2, the model is mainly composed of a fine-tuned LLM and stable diffusion. To align the LLM with the text encoder of stable diffusion, we used mapping layers for mapping and used a parameter-efficient adapter to fine-tune the LLM.

Firstly, after inputting the text, the model will use a router to determine whether the input instruction involves an image generation task. Specifically, we use the simple naive Bayes algorithm. We generated hundreds of instructions with GPT-4 as a corpus. If not generating images, it will use the LLM for normal conversation. At this time, we will not add an adapter to the LLM, and we will use all the parameters pre-trained by the LLM without any modification. If the model determines that the input prompt is asking for image generation, then we will add a trained adapter to the LLM and connect a mapping layer to map the output of the LLM to the same input dimension required by the diffusion model. At this time, the LLM serves as the text encoder of the diffusion model, inputs the text conditioning into the diffusion model, and guides the diffusion model to generate the picture required by the prompt. Through such operations, we can use the same LLM for both chatting and text-to-image tasks. The parameters of the adapter and mapping layers are less than 1% of the entire model, which greatly reduces the training cost. In addition, by using different adapters, the LLM can also serve as the text encoder of models that generate audio, video, etc. Due to time and computational constraints, we were unable to demonstrate such results here, which is our future work.

Implementation. In this model, we chose LLaMA2-chat-7B as the LLM(Touvron et al. 2023). Before entering the mapping layers, we extract the features of the last hidden layer of LLaMA2-chat-7B, which is 4096-dimensional. Then we directly use multiple linear layers to map the 4096-dimensional features to 768 dimensions, which serves as the input to the diffusion model. The hidden dimension of the linear layer is set to 512. For the adapter, we use RepAdapter(Luo et al. 2023) on all linear layers. This is because RepAdapter performs well in the task of fine-tuning the LLM, and it can add no burden to the inference process through reparameterization. Finally, considering the ease of training and the amount of reference experience, we use stable diffusion v1.4 as the diffusion model. The output of the LLM after mapping serves as the text condition input to stable diffusion v1.4. The size of the generated image is 512*512.

Training. In response to the characteristics of the model, we propose a two-stage training strategy: the mapping layer stage and the adapter stage. Let's use c to represent the input text.

In the mapping layer stage, we only train the linear mapping layers. We make the features of the LLM after mapping close to the result of the CLIP-encoded text originally used by stable diffusion. So we propose caption loss:

$$L_{cap} = MSE(Mapping(LLM(c)), CLIP(c)) \quad (1)$$

The training at this stage can be seen as a form of distillation, where we make the output of the LLM similar to CLIP after mapping through the mapping layers. After this stage of training, the model can already generate decent images, but there is still a certain gap compared to stable diffusion.

In the adapter stage, we insert adapters into all linear layers of the LLM (including mapping layers). In addition to freezing all parameters of the LLM as before, this stage also freezes the parameters of the mapping layers. The adapter stage uses two losses. In addition to the caption loss shown in Equation (1), we also use the diffusion loss L_{dif} in this stage, which is used in the diffusion model's u-net to predict noise.

Therefore, the total loss of the adapter stage can be expressed as:

$$L_{total} = L_{cap} + \lambda * L_{dif} \quad (2)$$

Where L_{cap} is the caption loss and L_{dif} is the diffusion loss. And λ denotes the coefficient to balance these two losses. In practice, we set λ to 0.1.

After two stages of training, the image generation effect of the model is comparable to that of stable diffusion. During the entire training process, we only used 3 Nvidia A800 GPUs and trained for two weeks. The trainable parameters in the entire model account for less than 1% of the entire model, and due to the use of reparameterization technology, adding an adapter will not have any impact on the inference speed. Using such a simple and efficient training method, in the future, we can use just one LLM to simultaneously accomplish tasks such as text-to-audio and text-to-video, and the training cost is only a few tens of Nvidia A800 GPU days.

Experiments

Firstly, we have empirically demonstrated that our model performs comparably to stable diffusion in the task of text-to-image generation. Additionally, we conducted a series of ablation experiments to validate the rationality of our design.

Text-to-image

Dataset. The Microsoft COCO (Common Objects in Context) dataset(Lin et al. 2014) is a widely recognized benchmark for evaluating the performance of text-to-image synthesis models. It contains over 200,000 labeled images spanning 80 categories, providing a diverse and comprehensive dataset for training and validating models.

Each image in the COCO dataset has at least five different captions, providing rich textual descriptions that correlate with the visual content. This makes it ideal for text-to-image

Settings	FID	CLIP-Sim	User Study
SD v1.4	**15.17**	**0.3137**	58.33%
Ours	**15.21**	**0.3021**	41.67%
w/o L_{dif}	16.02	0.3013	-
w/o Adapter	20.32	0.2929	-
w/o Adapter and L_{dif}	16.49	0.2995	-

Table 1: The results of our model and the ablation experiments are presented. Notably, the results of our model show a minimal difference compared to the original version of stable diffusion.

tasks, as it allows models to learn the complex relationships between textual inputs and their corresponding visual representations.

We use the COCO dataset to train our text-to-image synthesis model. The dataset's diversity, in terms of visual content and associated textual descriptions, enables our model to learn to generate a wide range of images from textual descriptions. The large number of images ensures that our model encounters a variety of scenarios during training, enhancing its ability to generalize to unseen inputs during testing.

We also use the COCO dataset for validation and testing, allowing us to quantitatively evaluate our model's performance. This ensures that our model generates visually appealing images that accurately reflect the provided textual descriptions.

Metrics. In the evaluation of text-to-image synthesis models, two key metrics are often employed: the Frechet Inception Distance (FID)(Heusel et al. 2017) and CLIP-similarity.

Frechet Inception Distance (FID) is a metric used to quantitatively evaluate the quality of generated images by measuring the statistical similarity between the distribution of generated and real images. Lower FID scores suggest that the generated images are of higher quality and more realistic. The FID score is computed using features extracted from an intermediate layer of the Inception network, which are assumed to follow a Gaussian distribution. The Frechet distance between these two Gaussians is then calculated, providing a measure of the distance between the distributions of real and generated images.

CLIP-similarity is a metric designed to assess the semantic alignment between a generated image and the input text. It utilizes the CLIP (Contrastive Language-Image Pretraining) model, which is trained to understand images and text in a joint embedding space. The CLIP similarity score is computed as the cosine similarity between the CLIP embeddings of a generated image and the corresponding input text. A higher CLIP-similarity score indicates better semantic alignment between the generated image and the input text.

In our experiments, we use both FID and CLIP similarity to evaluate the performance of our text-to-image synthesis model. The FID score provides a measure of the visual quality of our generated images, while the CLIP-similarity score assesses the semantic alignment between our generated images and the input text. Together, these two metrics provide a comprehensive evaluation of our model's performance.

Results. As shown in Table 1, our model's image generation capability is very close to the original version of stable diffusion, with only a slight increase in FID by 0.4 and a decrease in CLIP similarity by 0.0124. To ensure that these metrics adequately measure the image generation performance of our model, we also conducted a user study. We generated images from 100 prompts for both models and asked participants, who were unaware of the models' identities, to choose which model generated better images based on text-image matching and image realism. The final results showed that the generation effects of the two models are comparable. According to the feedback of the participants, our model mainly has a slight gap in image-text matching, which is also consistent with the quantitative results.

Ablation Study

To validate the rationality and non-redundancy of our model design, we conducted a series of ablation experiments, as shown in Table 1. Firstly, if diffusion loss is not used in both stages of training, the results obtained are slightly inferior to those using diffusion loss, as shown in the third row of the table. The most interesting part is the last two rows of the table. If we remove the adapter but keep the loss unchanged, the image generation effect will be very poor. However, if the adapter is not used and diffusion loss is also not used, the image generation effect can still be acceptable. We speculate that without the use of the adapter and only using the mapping layer, it may be difficult to optimize the pixel-level loss of diffusion loss, because the representation capability of the mapping layer is too weak. Therefore, diffusion loss must be used in conjunction with the adapter.

Conclusion

This paper proposes that by using only the adapter and mapping layers, with less than 1% of the parameters fine-tuned, the Language-to-Latent Model (LLM) can be combined with the diffusion model. Experiments show that its text-to-image generation performance is comparable to that of stable diffusion. More importantly, using this method, models for text-to-video and text-to-audio can also be integrated. By fine-tuning only one LLM, it is possible to connect text with many modalities. Moreover, the training cost is low, and the practical use effect is good.

References

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

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*.

Chung, H. W.; Hou, L.; Longpre, S.; Zoph, B.; Tay, Y.; Fedus, W.; Li, E.; Wang, X.; Dehghani, M.; Brahma, S.; et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34: 8780–8794.

Ding, M.; Yang, Z.; Hong, W.; Zheng, W.; Zhou, C.; Yin, D.; Lin, J.; Zou, X.; Shao, Z.; Yang, H.; et al. 2021. Cogview: Mastering text-to-image generation via transformers. *Advances in Neural Information Processing Systems*, 34: 19822–19835.

Ding, M.; Zheng, W.; Hong, W.; and Tang, J. 2022. Cogview2: Faster and better text-to-image generation via hierarchical transformers. *Advances in Neural Information Processing Systems*, 35: 16890–16902.

Esser, P.; Rombach, R.; and Ommer, B. 2021. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12873–12883.

Fan, W.-C.; Chen, Y.-C.; Chen, D.; Cheng, Y.; Yuan, L.; and Wang, Y.-C. F. 2023. Frido: Feature pyramid diffusion for complex scene image synthesis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 579–587.

Girdhar, R.; El-Nouby, A.; Liu, Z.; Singh, M.; Alwala, K. V.; Joulin, A.; and Misra, I. 2023. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15180–15190.

Gu, S.; Chen, D.; Bao, J.; Wen, F.; Zhang, B.; Chen, D.; Yuan, L.; and Guo, B. 2022. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10696–10706.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Ho, J.; Saharia, C.; Chan, W.; Fleet, D. J.; Norouzi, M.; and Salimans, T. 2022. Cascaded diffusion models for high fidelity image generation. *The Journal of Machine Learning Research*, 23(1): 2249–2281.

Huang, S.; Dong, L.; Wang, W.; Hao, Y.; Singhal, S.; Ma, S.; Lv, T.; Cui, L.; Mohammed, O. K.; Liu, Q.; et al. 2023. Language is not all you need: Aligning perception with language models. *arXiv preprint arXiv:2302.14045*.

Koh, J. Y.; Fried, D.; and Salakhutdinov, R. 2023. Generating images with multimodal language models. *arXiv preprint arXiv:2305.17216*.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023a. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*.

Li, K.; He, Y.; Wang, Y.; Li, Y.; Wang, W.; Luo, P.; Wang, Y.; Wang, L.; and Qiao, Y. 2023b. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*.

Li, Y.; Zhang, C.; Yu, G.; Wang, Z.; Fu, B.; Lin, G.; Shen, C.; Chen, L.; and Wei, Y. 2023c. StableLLaVA: Enhanced Visual Instruction Tuning with Synthesized Image-Dialogue Data. *arXiv preprint arXiv:2308.10253*.

Lin, J.; Men, R.; Yang, A.; Zhou, C.; Ding, M.; Zhang, Y.; Wang, P.; Wang, A.; Jiang, L.; Jia, X.; et al. 2021. M6: A chinese multimodal pretrainer. *arXiv preprint arXiv:2103.00823*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 740–755. Springer.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Luo, G.; Huang, M.; Zhou, Y.; Sun, X.; Jiang, G.; Wang, Z.; and Ji, R. 2023. Towards efficient visual adaption via structural re-parameterization. *arXiv preprint arXiv:2302.08106*.

Nichol, A.; Dhariwal, P.; Ramesh, A.; Shyam, P.; Mishkin, P.; McGrew, B.; Sutskever, I.; and Chen, M. 2021. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*.

Nichol, A. Q.; and Dhariwal, P. 2021. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, 8162–8171. PMLR.

OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1): 5485–5551.

Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; and Chen, M. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2): 3.

Ramesh, A.; Pavlov, M.; Goh, G.; Gray, S.; Voss, C.; Radford, A.; Chen, M.; and Sutskever, I. 2021. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, 8821–8831. PMLR.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Saharia, C.; Chan, W.; Saxena, S.; Li, L.; Whang, J.; Denton, E. L.; Ghasemipour, K.; Gontijo Lopes, R.; Karagol Ayan, B.; Salimans, T.; et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35: 36479–36494.

Shen, Y.; Song, K.; Tan, X.; Li, D.; Lu, W.; and Zhuang, Y. 2023. Hugginggpt: Solving ai tasks with chatgpt and its friends in huggingface. *arXiv preprint arXiv:2303.17580*.

Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, 2256–2265. PMLR.

Stiennon, N.; Ouyang, L.; Wu, J.; Ziegler, D.; Lowe, R.; Voss, C.; Radford, A.; Amodei, D.; and Christiano, P. F. 2020. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33: 3008–3021.

Su, Y.; Lan, T.; Li, H.; Xu, J.; Wang, Y.; and Cai, D. 2023. Pandagpt: One model to instruction-follow them all. *arXiv preprint arXiv:2305.16355*.

Su, Y.; Lan, T.; Liu, Y.; Liu, F.; Yogatama, D.; Wang, Y.; Kong, L.; and Collier, N. 2022. Language models can see: Plugging visual controls in text generation. *arXiv preprint arXiv:2205.02655*.

Tang, Z.; Yang, Z.; Zhu, C.; Zeng, M.; and Bansal, M. 2023. Any-to-Any Generation via Composable Diffusion. *arXiv preprint arXiv:2305.11846*.

Taori, R.; Gulrajani, I.; Zhang, T.; Dubois, Y.; Li, X.; Guestrin, C.; Liang, P.; and Hashimoto, T. B. 2023. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models. https://crfm. stanford. edu/2023/03/13/alpaca. html*, 3(6): 7.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Van Den Oord, A.; Vinyals, O.; et al. 2017. Neural discrete representation learning. *Advances in neural information processing systems*, 30.

Wu, C.; Yin, S.; Qi, W.; Wang, X.; Tang, Z.; and Duan, N. 2023. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*.

Yu, J.; Xu, Y.; Koh, J. Y.; Luong, T.; Baid, G.; Wang, Z.; Vasudevan, V.; Ku, A.; Yang, Y.; Ayan, B. K.; et al. 2022. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2(3): 5.

Zhang, A.; Fei, H.; Yao, Y.; Ji, W.; Li, L.; Liu, Z.; and Chua, T.-S. 2023a. Transfer visual prompt generator across llms. *arXiv preprint arXiv:2305.01278*.

Zhang, D.; Li, S.; Zhang, X.; Zhan, J.; Wang, P.; Zhou, Y.; and Qiu, X. 2023b. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. *arXiv preprint arXiv:2305.11000*.

Zhang, H.; Li, X.; and Bing, L. 2023. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*.

Zhang, Y.; Zhang, R.; Gu, J.; Zhou, Y.; Lipka, N.; Yang, D.; and Sun, T. 2023c. Llavar: Enhanced visual instruction tuning for text-rich image understanding. *arXiv preprint arXiv:2306.17107*.

Zhu, D.; Chen, J.; Shen, X.; Li, X.; and Elhoseiny, M. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.