TEAL: Tokenize and Embed ALL for Multi-modal Large Language Models

Anonymous ACL submission

Abstract

Despite Multi-modal Large Language Models (MM-LLMs) have made exciting strides recently, they are still struggling to efficiently model the interactions among multi-modal inputs and the generation in non-textual modalities. In this work, we propose *TEAL (Tokenize and Embed ALL)*, an approach to treat the input from any modality as a token sequence and learn a joint embedding space for all modalities. Specifically, for the input from any modality, *TEAL* firstly discretizes it into a token sequence with the off-the-shelf tokenizer and embeds the token sequence into a joint embedding space with a learnable embedding matrix. MM-LLMs just need to predict the multi-modal tokens autoregressively as conventional textual LLMs do. Finally, the corresponding de-tokenizer is applied to generate the output in each modality based on the predicted token sequence. With the joint embedding space, *TEAL* enables the frozen LLMs to perform both understanding and generation tasks involving non-textual modalities, such as image and audio. Thus, the textual LLM can just work as an interface and maintain its high performance in textual understanding and generation. Experiments show that *TEAL* achieves substantial improvements in multi-modal understanding, and implements a simple scheme for multi-modal generation.

1 Introduction

Recently, Multi-Modal Large Language Models (MM-LLMs), which perform understanding and generation tasks more than textual modalities, have made exciting strides and garnered significant attention for their potential in Artificial Intelligence Generated Content (AIGC) (Cao et al., 2023). MM-LLMs are considered a step closer to Artificial General Intelligence (AGI) (Goertzel and Pennachin, 2007; Fei et al., 2022) due to their provision of more user-friendly interfaces and their ability to perceive the world similarly to humans (Yin et al., 2023). Typically, there are two main different branches in the realm of constructing MM-LLMs: One branch aims to construct a 'real' multi-modal model by training the model with multi-modal data from scratch, without relying on the pre-trained textual LLMs (Borsos et al., 2023; Lu et al., 2022a; Barrault et al., 2023; Shukor et al., 2023; Chen et al., 2023c; Copet et al., 2023); The other branch takes the textual LLMs as the backbone and enables them to perform multi-modal understanding and generation tasks with instruction tuning.

With the rapid advancement of textual LLMs, researchers are keener on the second branch of approaches which empowers the pre-trained high-performance textual LLMs with multi-modal abilities. In this line, some typical works, such as BLIP-2 (Li et al., 2023), Flamingo (Alayrac et al., 2022), MiniGPT-4 (Zhu et al., 2023), LLama-Adapter (Gao et al., 2023; Zhang et al., 2023c), LLaVA (Liu et al., 2023b,a), SpeechGPT (Zhang et al., 2023a), involve employing adapters that align pre-trained encoders in other modalities to textual LLMs. As these works take the dense features from the pre-trained encoders as additional non-textual information, they cannot efficiently model the interactions among multi-modal inputs and falter in the nuanced art of generating non-textual content. To compensate for this deficiency in the non-textual generation, some efforts, such as visual-ChatGPT (Chen et al., 2023c), Hugging-GPT (Shen et al., 2023), Audio-GPT (Huang et al., 2023), Next-GPT (Wu et al., 2023b), and MiniGPT-5 (Zheng et al., 2023) have sought to amalgamate the textual LLMs with some external generation tools, e.g., Stable Diffusion (Rombach et al., 2022), DALL-E (Ramesh et al., 2021), Whisper (Radford et al., 2023). Unfortunately, these systems suffer from two critical challenges due to their complete pipeline architectures. First, the information transfer between different modules is entirely based on generated textual tokens, where the process may lose some multi-modal information and propagate errors (Wu et al., 2023b). Additionally, the external tools usually make the models complex and heavy, which consequently results in inefficient training and inference.

Based on the above observation, we conclude that the emerging challenges in the previous works are mainly raised by their non-unified processing of the multi-modal inputs, where they encode the non-textual inputs into a dense and high-level feature, but tokenize the textual input into a token sequence. The non-unified processing introduces an extra burden for LLMs to model the interaction between multi-modal inputs and generate the non-textual samples. In a nutshell, if we can tokenize the interleaved multi-modal input into a token sequence and align the non-textual token embedding into the textual embedding space, the original textual LLMs can be easily transformed to handle non-textual understanding and generation tasks with parameters tuned as little as possible.

In pursuit of this goal and inspired by the recent advancement of multi-modal tokenizers (Yu et al., 2023b; Chang et al., 2023; Peng et al., 2022; Borsos et al., 2023; Yu et al., 2023a), we propose *TEAL*, a token-in-token-out MM-LLM designed to seamlessly handle the token input and output in any combination of three modalities: text, image, and audio. Specifically, *TEAL* comprises three tiers. First, we tokenize the input from any modality into a token sequence with the off-the-shelf tokenizers, such as BEiT-V2 and a Whisper-based audio tokenizer. Second, we insert a non-textual embedding matrix and output matrix into an open-source textual LLM, which enables the textual LLM to process the non-textual inputs and outputs. To align the non-textual embedding matrices with their textual counterparts, we equip them with a projection layer. Third, the generated tokens are routed to the corresponding de-tokenizers, which transform the token sequences into samples in different modalities. We test the effectiveness and generality of our method by conducting extensive experiments on the modalities of text, image, and audio. We also make a deep investigation into the tokenizers in each modality, which is the core component of our method.

In summary, our contributions are three-fold:

1. We propose *TEAL*, an approach that treats the input from any modality as a token sequence and learns a joint embedding space for all modalities. *TEAL* introduces a simple way to enable the frozen LLMs to perform both understanding and generation tasks involving non-textual modalities.

2. We conduct extensive experiments on the non-textual modalities of image and audio. Experimental results show that *TEAL* achieves substantial improvements over previous works on multi-modal understanding and paves a simple way for the generation of non-textual modalities. To the best of our knowledge, this is the first work that successfully empowers the frozen LLM to perform tasks involving both the non-textual modalities of audio and image.

3. By testing versatile tokenizers for image and audio, we find that the tokenizer is the key to the performance of MM-LLMs. Our extensive experiments have identified a new research direction that devising a general semantic-aware tokenizer is very promising.

2 Related Work

2.1 MM-LLMs

Training a multi-modal large language model from scratch in an end-to-end manner incurs substantial costs. Therefore, most researchers choose to integrate multi-modal modules into existing text-based large language models, allowing these models to acquire multi-modal capabilities. One branch involves employing robust pre-trained vision or audio encoders to encode multi-modal information into features and subsequently align it with the feature space of an LLM (Dai et al., 2023; Chen et al., 2023a; Zhang et al., 2023b,c; Gao et al., 2023; Ling et al., 2023; Wu et al., 2023a; Hussain et al., 2023). For example, Flamingo (Alayrac et al., 2022) utilizes vision encoders to obtain a fixed number of visual tokens and use cross-attention layers to connect the pre-trained LLM layers. BLIP-2 (Li et al., 2023) utilizes a Q-Former as a bridge between the input image and the LLMs. LauraGPT (Chen et al., 2023b) uses a pre-trained Conformer-based encoder to extract continuous audio representations for the connected LLM. Furthermore, different projection layers are used to reduce the modality gap, such as a simple Linear Layer (Liu et al., 2023a) or a two-layer Multi-layer Perceptron (Zhang et al., 2023d). Moreover, LLaMa-Adapter (Zhang et al., 2023c; Gao et al., 2023) integrates trainable adapter modules into LLMs, enabling effective parameter tuning for the fusion of multi-modal information. Another branch involves using off-the-shelf expert models to convert images or speech into natural language in an offline manner, such as Next-GPT (Wu et al., 2023b), SpeechGPT (Zhang et al., 2023a) and AudioGPT (Huang et al., 2023).

Contrary to these works mentioned above, we tokenize the input from any modality into a token sequence and train a token-in-token-out MM-LLM designed to seamlessly handle the token input and output in any combination of three modalities: text, image, and audio. Gemini (Team et al., 2023) is our concurrent work which adopts a similar technical approach as ours.

2.2 Non-textual Discretization

In addition to directly integrating multi-modal modules or using offline expert models, there are also efforts focused on non-textual discretization, which employs tokenizers to convert continuous images or audio into token sequences. This way, all modalities share the same form as tokens, which can be better compatible with LLM. Next, we will introduce two mainstream methods of Non-textual discretization.

VQ-VAEs Vector Quantised Variational AutoEncoder (VQ-VAE) (Van Den Oord et al., 2017) is a seminal contribution in the field of non-textual tokenization, which incorporates vector quantization (VQ) to learn discrete representations and converts images into a sequence of discrete codes. In the vision domain, VQGAN (Esser et al., 2021) follows the idea, using a codebook to discretely encode images, and employs Transformer as the encoder. ViT-VQGAN (Yu et al., 2021) introduces several enhancements to the vanilla VQGAN, encompassing architectural modifications and advancements in codebook learning. BEiT-V2 (Peng et al., 2022) proposes Vector-quantized Knowledge Distillation (VQ-KD) to train a semantic-rich visual tokenizer by reconstructing high-level features from the teacher model. Ge et al. (2023) propose SEED and claims two principles for the tokenizer architecture and training that can ease the alignment with LLMs. Yu et al. (2023a) introduce SPAE, which can convert between raw pixels and lexical tokens extracted from the LLM's vocabulary, enabling frozen LLMs to understand and generate images or videos. For the audio, Dieleman et al. (2018) utilize autoregressive discrete autoencoders (ADAs) to capture correlations in waveforms. Jukebox (Dhariwal et al., 2020) uses a multi-scale VQ-VAE to compress music to discrete codes and model those using autoregressive Transformers, which can generate music with singing in the raw audio domain. SoundStream (Zeghidour et al., 2021) employs a model architecture composed of a fully convolutional encoder/decoder network and adopts a Residual Vector Quantizer (RVQ) to project the audio embedding in a codebook of a given size. Défossez et al. (2022), Jiang et al. (2022) also adopt RVQ to quantize the output of the encoder.

Clustering Except for those methods that use trained specialized vector quantization (VQ) modules as tokenizers, some works (Lakhotia et al., 2021; Kharitonov et al., 2022) apply the clustering algorithms to the features, and the cluster indices are directly used as the discrete tokens for speech. The cluster approach typically relies on self-supervised learning models, such as HuBERT (Hsu et al., 2021), W2V-BERT (Chung et al., 2021; Borsos et al., 2023), USM (Zhang et al., 2023e; Rubenstein et al., 2023), which are trained for discrimination or masking prediction and maintain semantic information of the speech. Compared with neural VQ-based tokenizers, the clustering-based approach provides enhanced flexibility as it can be applied to any pre-trained speech model without altering its underlying model structure.

3 Method

The main goal of this paper is to enable the frozen textual LLMs to model sequences consisting of multi-modal discrete tokens. Thus, the textual LLMs obtain the ability to perform both understanding and generation tasks involving non-textual modalities and maintain their strong abilities in text. The main architecture of our method is illustrated in Figure 1. Firstly, we discretize the interleaved multi-modal input into a token sequence with the off-the-shelf tokenizers. Then, an open-source textual LLM is used to model the input and output token sequence by aligning the textual and non-textual embedding space. Finally, the corresponding off-the-shelf decoder is utilized to generate the output in each modality. In the remainder of this section, we will describe the model architecture in Subsection 3.1. The tokenizer and de-tokenizer for non-textual modalities we used in this paper will be

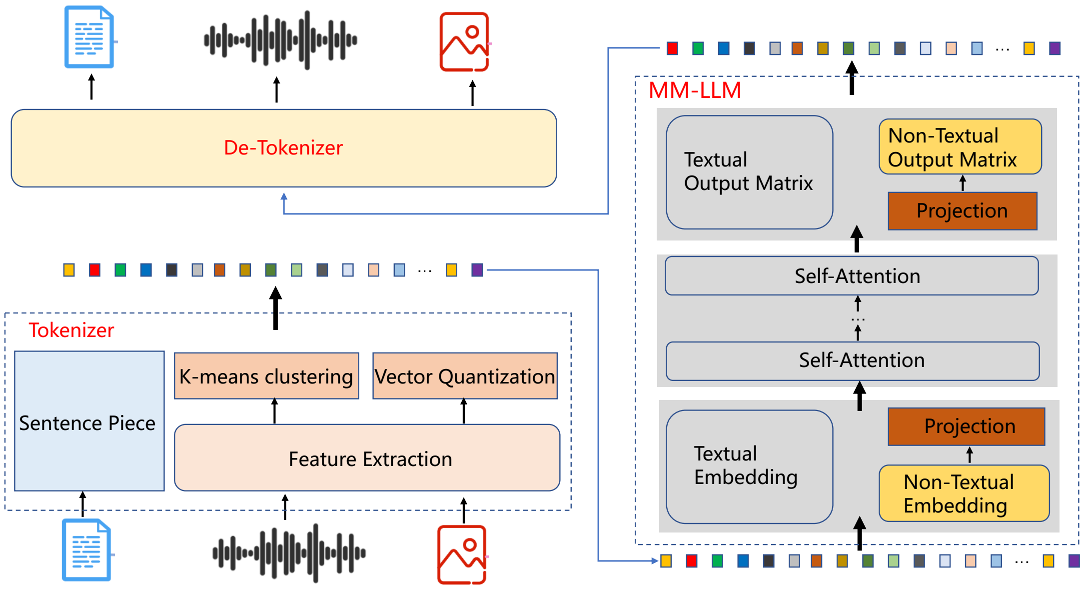


Figure 1: The main architecture of *TEAL*. The modules in MM-LLM denoted with the color gray make up the original textual LLM and most of them are frozen during training.

presented in Subsection 3.2. Finally, we propose our two-stage training strategies in Subsection 3.3.

3.1 Model Architecture

TEAL is a general method that can be applied to any open-source LLMs. In this paper, the proposed MM-LLM takes the most popular open-source textual LLM, i.e., LLaMA, as the backbone, which makes it easy to compare fairly with previous works. To support the modeling of non-textual tokens, the MM-LLM also incorporates a non-textual embedding layer and a non-textual output layer. Two projection layers are applied after the non-textual embedding layer and before the output layer separately, which mainly serve two purposes: 1) make the output dimension of textual and non-textual embedding the same; 2) align the non-textual embedding with the textual embedding space. To ease the training process and solve the cold-start problem, we initialize the non-textual embedding and output matrix with the codebook of the tokenizer, which will be described in Subsection 3.2 in detail.

3.2 Tokenize and De-Tokenize

Tokenization is a very popular technique in the area of natural language processing, which is usually used as a tool to split the input sentence into the granularity of sub-words. Most of the existing textual LLMs take the sentence piece as the tokenizer for its universal processing of multi-lingual texts. The de-tokenization for the sentence piece is very simple, which just works as a function to replace the meta-symbol '_' with the whitespace. Recently, tokenization (or denoted as discretization) in non-textual modalities has gained much attention and achieved substantial improvements, which makes it possible to build a fully token-in-token-out MM-LLM. The most widely used methods are VQ-VAE and k-means clustering. In this paper, we take the encoder of the VQ-VAE models and the k-means clustering as the tokenizers for the image and audio respectively. The decoders of the VQ-VAE models are taken as the de-tokenizers for the image and audio. For the image, we test three different tokenizers, namely DALL-E (Ramesh et al., 2021), VQ-GAN (Esser et al., 2021) and BEiT-V2 (Peng et al., 2022). For the audio, we apply K-means Clustering on the intermediate features of the following typical models, and the cluster indices are directly used as the discrete tokens for speech. We test two different tokenizers for audios, such as HuBERT (Hsu et al., 2021) and Whisper (Radford et al., 2023). We present detailed descriptions of these tokenizers and test their effects on the final performance in Section 5.1.

3.3 Two-stage Supervised Finetuning

The proposed *TEAL* model is initialized with the open-source textual LLM. To obtain the understanding and generation ability in non-textual modalities and maintain its high performance in textual modality, we propose a two-stage supervised fine-tuning that trains the model with parameters tuned as little as possible. In the following, we denote the two stages of supervised fine-tuning as pre-training and fine-tuning separately.

Pre-training The goal of the pre-training is to align the non-textual and textual embedding space by tuning the projection layer. Specifically, we freeze all parameters in the MM-LLM except the parameter of the two projection layers. We generate the training samples from the vision-language and audio-language pairs with very simple prompts. Taking the vision-language pair as an example, we generate two training samples from each vision-language pair with the following format:

The image and text pair:[img][text]

The text and image pair:[text][img]

Fine-tuning In the stage of fine-tuning, we process the corpus of downstream tasks as the prompt format in (Zhang et al., 2023c). For each task, we use the GPT4 to generate 10 different prompts.[1] We freeze the parameters of the textual LLM and tune all parameters related to the non-textual modalities. Following (Zhang et al., 2023c), we apply the bias-norm tuning where the bias and norm parameters are inserted in each layer to enhance the fine-tuning performance. We also tested LoRA tuning (Hu et al., 2021), but we did not obtain further improvement.

4 Experiments

We mainly test our method on understanding tasks involving non-textual modalities. To show the non-textual generation abilities, we will show our performance on the text-to-image generation.

4.1 Setup

For image-related understanding tasks, we test our method on CoCo-caption and science-QA. Additionally, we test our method's ability to understand speech information in the tasks of automatic speech recognition and speech translation. In the following experiments, we use BEiT-V2 and Whisper as the tokenizers for the image and audio understanding respectively. The embeddings and output matrix for non-textual modalities are initialized with the codebook embeddings of the corresponding tokenizers. The model is implemented based on the codebase of LLaMA-Adapter (Gao et al., 2023).[2] If there is no specific explanation, all models are trained with two-stage supervised fine-tuning on 8 A100 GPUs, and the main hyper-parameters are set the same with LlaMA-Adapter. During the pre-training phase, we did not introduce any additional data apart from the training data for the tasks mentioned above. During fine-tuning, we also include the corpus of alpaca to enhance the model's ability on text understanding (Taori et al., 2023). All the data for different tasks are processed into a unified format and trained without explicitly differentiating between the tasks during the training process. Following (Gao et al., 2023), we adopt top-p sampling as the default decoding method with a temperature of 0.1 and a top-p of 0.75.

4.2 Main Results on Image Understanding

CoCo-Caption Image captioning is the task of generating descriptive captions for images. We utilize all image-caption pairs from the coco2014 dataset (Chen et al., 2015), which contains 83K images for training. As there are at least five captions for each image in the COCO2014 dataset, we can construct at least five training examples for each image by pairing the image with its all captions respectively. For a fair comparison, we report the CIDER, BLEU-4 on the Karpathy test split, which is evaluated with the official toolkit, pycocoeval.[3] The result is presented in Table 1. From Table 1, we can find that *TEAL* achieves substantial improvements compared to the baseline of LLaMA-Adapter v2, which applies a frozen vision encoder to incorporate the vision information. Specifically, we achieve 1.3 and 5.8 points improvement on the metrics of BLEU-4 and CiDER respectively. Additionally, compared to the models that trained with large-scale corpora, such as the BLIP and BLIP2, *TEAL* further narrows the performance gap without additional pre-training corpus. The cases on the valid set are shown in Appendix B. We can find that *TEAL* can understand the content of images well and describe the details of the images clearly.

ScienceQA ScienceQA (Lu et al., 2022b) is collected from elementary and high school science curricula and contains 21,208 multimodal multiple-choice science questions. Out of the questions in ScienceQA, 10,332 (48.7%) have an image context, 10,220 (48.2%) have a text context, and 6,532

[1]For details of the prompt format, we refer the readers to the Appendix A.

[2]https://github.com/Alpha-VLLM/LLaMA2-Accessory

[3]https://github.com/cocodataset/cocoapi

Model	Data Scale		COCO Caption	
	PT	FT	CiDER	BLEU-4
LlaMA-Adapter v2 (Gao et al., 2023)	0	0.6M	122.2	36.2
BLIP (Li et al., 2022)	14M	0.6M	136.7	40.4
BLIP2 (Li et al., 2023)	129M	0.6M	145.3	43.7
TEAL (Ours)	0	0.6M	128.0	37.5

Table 1: Model performance on the COCO2014 test set. The results of the baselines are cited from their papers directly.

Method	**Subject**			**Conext Modality**			**Grade**		Average
	NAN	SOC	LAN	TXT	IMG	NO	G1-6	G7-12	
LLaMA-Adapter	84.37	88.30	84.36	83.72	80.32	86.90	85.83	84.05	85.19
Human	90.23	84.97	87.48	89.60	87.50	88.10	91.59	82.42	88.40
GPT-3.5	74.64	69.74	76.00	74.44	67.28	77.42	76.80	68.89	73.97
GPT-3.5 w/ COT	75.44	70.87	78.09	76.48	67.43	79.93	78.23	69.68	75.17
MM-COT$_{base}$	87.52	77.17	85.82	87.88	82.90	86.83	84.65	85.37	84.91
MM-COT$_{large}$	95.91	82.00	90.82	95.26	88.80	92.89	92.44	90.31	91.68
LLaVA-7B	-	-	-	-	-	-	-	-	89.84
LLaVA-13B	90.36	95.95	88.00	89.49	88.00	90.66	90.93	90.90	90.92
TEAL (Ours)	88.91	92.98	86.31	86.06	82.85	88.97	86.56	84.80	87.18

Table 2: Results on the ScienceQA test set. For the baselines, we directly cite the results from their papers.

Model	clean	other
LauraGPT Discrete (Chen et al., 2023b)	9.1	24.0
Whisper$_{small}$ (Radford et al., 2023)	4.4	8.4
Whisper$_{large}$ (Radford et al., 2023)	2.7	5.2
Whisper$_{small}$ + LLaMa-Adapter	23.2	25.9
TEAL (Ours)	5.1	11.1

Table 3: Results on the LibriSpeech test-clean and test-other set.

Model	WER
HuBERT$_{large}$ (Hsu et al., 2021)	31.77
Whisper$_{small}$ (Radford et al., 2023)	18.8
Whisper$_{small}$ + LLaMa-Adapter	26.96
TEAL (Ours)	24.22

Table 4: Results on the CoVoST 2 ASR test set.

(30.8%) have both. ScienceQA has rich domain diversity across 3 subjects, 26 topics, 127 categories, and 379 skills, and the benchmark dataset is split into training, validation, and test splits with 12,726, 4,241, and 4,241 examples, respectively. The main baseline that can be used to make a fair comparison with our method is the LLaMA-Adapter (Zhang et al., 2023c). We also cite the results of two representation methods (GPT-3.5 and GPT-3.5 w/ COT) (Lu et al., 2022b), one multi-modal COT method (MM-COT) (Zhang et al., 2023f), human evaluation (Lu et al., 2022b), and LLaVA (Liu et al., 2023b) which tunes the full parameters of the vicuna with large-scale multi-modal pre-training corpus. Table 2 presents the experimental results. As shown in Table 2, we can find *TEAL* achieves about 2 points improvement on average compared to the baseline of LLaMA-Adapter.

4.3 Main Results on Audio Understanding

We conduct audio experiments on the Automatic Speech Recognition (ASR) and Automatic Speech Translation (AST) tasks. The former is capable of transcribing spoken language into written text, while the latter translates speech from one language to text in another language. The audio tokenizer was implemented by applying k-means clustering on the 11th layer of Whisper$_{small}$.[4] The number of cluster centers is set as 8,192 and the effect of the number of cluster centers will be investigated in Appendix C. While training and inference, the audio and the corresponding prompt will be processed into token sequences and fed into the MM-LLM directly. For a fair comparison, our main baseline is also implemented based on LLaMa-Adapter and Whisper$_{small}$, where the Whisper$_{small}$ is utilized as an encoder to extract the dense audio fea-

[4]We tested different layers of Whisper$_{small}$ and obtained the best performance on 11th layer.

Model	BLEU
Transfomer (Wang et al., 2020)	25.4
LauraGPT Discrete (Chen et al., 2023b)	5.0
Whisper$_{small}$ + LLaMa-Adapter	20.2
TEAL (Ours)	26.4

Table 5: Results on the CoVoST AST test set.

Model	COCO Caption		ScienceQA (ave.)
	CiDER	BLEU-4	
DALLE	110.8	23.9	77.12
VQGAN	117.5	26.1	79.56
BEiT-V2	**130.1**	**37.6**	**88.00**

Table 6: The performance of different tokenizers on the validation sets of the COCO2014 and ScienceQA. We keep all parameters and data the same and only vary the tokenizers.

Tokenizer	LLM	WER
W2V-BERT	PaLM-8B	50.1
USM-v1	PaLM-8B	40.2
USM-v2	PaLM-8B	22.3
HuBERT	LLaMa-7B	56.2
Whisper$_{small}$	LLaMa-7B	24.2

Table 7: The performance of different tokenizers on the validation set of the CoVoST 2. We directly cite the results for AudioPalm from their paper.

tures from the raw audio waves. The parameters of Whisper$_{small}$ are kept frozen during training. We use the default adapter architecture to integrate the audio features into the MM-LLM.

LibriSpeech We conduct ASR experiments on LibriSpeech (Panayotov et al., 2015) dataset, which consists of 281,241 training samples, 2,703 dev-clean samples, 2,864 dev-other samples, 2,621 test-clean samples, and 2,940 test-other samples. We use the word error rate (WER) as the metric. As Table 3 shows, *TEAL* significantly outperforms Whisper$_{small}$ + LLaMa-Adapter which extracts continuous audio representations for LLM. We noticed that *TEAL* did not outperform Whisper, and there are two main reasons for this. Firstly, Whisper is an expert model in the ASR field and has been exposed to over 600,000 hours of audio data for training, while TEAL has only been exposed to less than 2,000 hours of audio data. Secondly, Whisper specializes in ASR, whereas *TEAL* can simultaneously support both ASR and AST tasks.

CoVoST 2 ASR CoVoST 2 (Wang et al., 2020) ASR English dataset contains 232,976 audio-text training pairs, 15,532 validation pairs, and 15,532 test pairs. As Table 4 shows, combining an audio tokenizer makes LLM possess better multi-modal understanding ability than explicitly integrating an audio encoder, with a WER score improvement of 2.74. This may be because having modalities in the same token format makes it easier to integrate multi-modal information for LLM.

CoVoST 2 AST We evaluate the AST performance on CoVoST 2 (En $\rightarrow$ Zh) dataset, which consists of 289,430/15,531/15,531 train/dev/test samples. Table 5 shows the results. Compared to the baseline incorporating continuous features, *TEAL* achieved a 6-point improvement.

4.4 Image Generation

Following (Yu et al., 2023a), we show several text-to-image generation examples on the MNIST dataset (Deng, 2012) in Figure 2. Different from (Yu et al., 2023a), we do not use any prompt example for in-context learning. As the BEiT-V2 is not good at image reconstruction, we apply the VQGAN as the tokenizer for image generation.[5] From Figure 2, we can find that *TEAL* empowers the frozen textual LLM with the ability to generate the image following the prompt query. We also test with complex questions requiring mathematical reasoning or common sense knowledge, and the model can give the right responses. These results show that *TEAL* not only learns how to generate non-textual content but also maintains its previous ability in textual understanding. We notice that the quality of the generated image is not so perfect, and we leave the work of polishing the quality of generated images in the next version.

5 Analysis and Discussion

5.1 Different Tokenizers

We show how the tokenizer affects the performance by testing different tokenizers. Results for the image are shown in Table 6. We find that different tokenizers result in significant differences in the final performance, and BEiT-V2 achieves the best result. Compared to the baseline of VQ-GAN, BEiT-v2 achieves 11.5 BLEU points improvement on the task of COCO-caption and 8.5 accuracy points on ScienceQA. The significant performance gap highlights the importance of the tokenizer. We speculate that the main reason for BEiT-v2 achieving such a significant advantage is that BEiT-v2 has acquired

[5]This is because the BEiT-V2 is not trained to reconstruct the image but to recover the prediction of its teacher model.

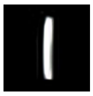
Prompt	Generation
### Instruction:\n\nPlease generate handwritten images corresponding to the input.\n\n ###Input:\n\nan image of 0	
### Instruction:\n\nPlease generate handwritten images corresponding to the input.\n\n ###Input:\n\nan image of the last digit of 3 plus 8	
### Instruction:\n\nPlease generate handwritten images corresponding to the input.\n\n ###Input:\n\nan image of the number of the continents in the world	
### Instruction:\n\nPlease generate handwritten images corresponding to the input.\n\n ###Input:\n\nan image of the number of the square of 3	

Figure 2: Some examples of the text-to-image generation on MNIST test set. We test with both simple and complex questions for the proposed *TEAL*.

Model	COCO Caption CiDER	BLEU-4
TEAL (Ours)	**130.1**	**37.6**
w/o 1st-stage finetuning	127.8	35.4
w/o embedding initialization	129.1	36.2
w/o bias-norm tuning	126.9	35.7

Table 8: Ablation study on *TEAL*. 'w/o 1st-stage finetuning' indicates that the model is trained with the 2nd-stage finetuning directly. 'w/o embedding initialization' means that we initialize the word embedding and output matrix randomly. 'w/o bias-tuning' means that the parameters of bias and norm are not added during the 2nd stage finetuning.

much semantic information during its pre-training, and the semantic information in the tokenizer is crucial for aligning different modalities.

We have similar observations in the modality of audio. In addition to Hubert and Whisper, we also introduce the results of AudioPaLM (Rubenstein et al., 2023) on some tokenizers based on non-open models (W2V-BERT (Chung et al., 2021), USM-v1 and v2 (Zhang et al., 2023e)) for a comprehensive comparison. The results are shown in Table 7. Both the results of AudioPaLM and *TEAL* demonstrate that the tokenizer has a significant impact on performance. Constructing a high-performance tokenizer is a very promising future work.

5.2 Ablation Study

To investigate the significance of each module in our model and method, we conduct an ablation study by training multiple versions of our model with some missing components, i.e., the 1st-stage finetuning, the embedding initialization, and the bias-norm tuning. We report the performance on the validation sets in Table 8. From Table 8, we can find that the best performance is obtained with the simultaneous use of all the tested components. The most critical components are the bias-norm tuning and the 1st-stage finetuning, which shows that the training strategies need to be carefully devised to ensure high performance. A surprising phenomenon is that when we randomly initialize the word embedding ('w/o embedding initialization' in Table 8), we do not observe a significant performance decrease. This result suggests that it is the way the tokenizer discretizes the image, rather than the word embedding preserved in the tokenizer, critical to the final performance. The reason why random initialization causes a certain degree of performance decrease is likely due to the relatively small size of the training data. We speculate that when the amount of training data reaches a certain level, the performance gap may disappear.

6 Conclusion and Future work

In this paper, we propose *TEAL*, an approach to training a fully token-in-token-out MM-LLM by treating the input from any modality as a token sequence and learning a joint embedding space for all modalities. *TEAL* empowers the frozen textual LLM with the ability to perform understanding and generation involving non-textual modalities. Extensive experiments show that, compared to the baseline models which integrate non-textual encoders, our approach achieves superior performance on non-textual understanding tasks, and paves a simple way for non-textual generation.

Limitations

Our approach relies on tokenizers for different modalities, and our experimental results show that tokenizers have a significant impact on overall performance. However, due to the lack of a universal tokenizer that performs well for both understanding and generation tasks, we are forced to use different tokenizers for each task, resulting in increased model complexity and modeling difficulties. This has become a bottleneck for the performance of our approach. To address this issue, one possible solution is to construct and train a universal tokenizer that supports both understanding and generation for different modalities. However, there are still many challenging problems that need to be resolved in this area.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736.

Loïc Barrault, Yu-An Chung, Mariano Cora Meglioli, David Dale, Ning Dong, Paul-Ambroise Duquenne, Hady Elsahar, Hongyu Gong, Kevin Heffernan, John Hoffman, et al. 2023. Seamlessm4t-massively multilingual & multimodal machine translation. *arXiv preprint arXiv:2308.11596*.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, et al. 2023. Audiolm: a language modeling approach to audio generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Yihan Cao, Siyu Li, Yixin Liu, Zhiling Yan, Yutong Dai, Philip S Yu, and Lichao Sun. 2023. A comprehensive survey of ai-generated content (aigc): A history of generative ai from gan to chatgpt. *arXiv preprint arXiv:2303.04226*.

Xuankai Chang, Brian Yan, Kwanghee Choi, Jeeweon Jung, Yichen Lu, Soumi Maiti, Roshan Sharma, Jiatong Shi, Jinchuan Tian, Shinji Watanabe, et al. 2023. Exploring speech recognition, translation, and understanding with discrete speech units: A comparative study. *arXiv preprint arXiv:2309.15800*.

Feilong Chen, Minglun Han, Haozhi Zhao, Qingyang Zhang, Jing Shi, Shuang Xu, and Bo Xu. 2023a. X-llm: Bootstrapping advanced large language models by treating multi-modalities as foreign languages.

Qian Chen, Yunfei Chu, Zhifu Gao, Zerui Li, Kai Hu, Xiaohuan Zhou, Jin Xu, Ziyang Ma, Wen Wang, Siqi Zheng, et al. 2023b. Lauragpt: Listen, attend, understand, and regenerate audio with gpt. *arXiv preprint arXiv:2310.04673*.

Xi Chen, Xiao Wang, Lucas Beyer, Alexander Kolesnikov, Jialin Wu, Paul Voigtlaender, Basil Mustafa, Sebastian Goodman, Ibrahim Alabdulmohsin, Piotr Padlewski, Daniel Salz, Xi Xiong, Daniel Vlasic, Filip Pavetic, Keran Rong, Tianli Yu, Daniel Keysers, Xiaohua Zhai, and Radu Soricut. 2023c. Pali-3 vision language models: Smaller, faster, stronger.

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*.

Yu-An Chung, Yu Zhang, Wei Han, Chung-Cheng Chiu, James Qin, Ruoming Pang, and Yonghui Wu. 2021. W2v-bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 244–250. IEEE.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. 2023. Simple and controllable music generation. *arXiv preprint arXiv:2306.05284*.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. 2022. High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*.

Li Deng. 2012. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE signal processing magazine*, 29(6):141–142.

Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. 2020. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*.

Sander Dieleman, Aaron van den Oord, and Karen Simonyan. 2018. The challenge of realistic music generation: modelling raw audio at scale. *Advances in neural information processing systems*, 31.

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

Nanyi Fei, Zhiwu Lu, Yizhao Gao, Guoxing Yang, Yuqi Huo, Jingyuan Wen, Haoyu Lu, Ruihua Song, Xin Gao, Tao Xiang, et al. 2022. Towards artificial general intelligence via a multimodal foundation model. *Nature Communications*, 13(1):3094.

Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, et al. 2023. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*.

Yuying Ge, Yixiao Ge, Ziyun Zeng, Xintao Wang, and Ying Shan. 2023. Planting a seed of vision in large language model. *arXiv preprint arXiv:2307.08041*.

Ben Goertzel and Cassio Pennachin. 2007. *Artificial general intelligence*, volume 2. Springer.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models.

Rongjie Huang, Mingze Li, Dongchao Yang, Jiatong Shi, Xuankai Chang, Zhenhui Ye, Yuning Wu, Zhiqing Hong, Jiawei Huang, Jinglin Liu, et al. 2023. Audiogpt: Understanding and generating speech, music, sound, and talking head. *arXiv preprint arXiv:2304.12995*.

Atin Sakkeer Hussain, Shansong Liu, Chenshuo Sun, and Ying Shan. 2023. Mugen: Multi-modal music understanding and generation with the power of large language models. *arXiv preprint arXiv:2311.11255*.

Xue Jiang, Xiulian Peng, Huaying Xue, Yuan Zhang, and Yan Lu. 2022. Cross-scale vector quantization for scalable neural speech coding.

Eugene Kharitonov, Ann Lee, Adam Polyak, Yossi Adi, Jade Copet, Kushal Lakhotia, Tu Anh Nguyen, Morgane Riviere, Abdelrahman Mohamed, Emmanuel Dupoux, and Wei-Ning Hsu. 2022. Text-free prosody-aware generative spoken language modeling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8666–8681, Dublin, Ireland. Association for Computational Linguistics.

Kushal Lakhotia, Eugene Kharitonov, Wei-Ning Hsu, Yossi Adi, Adam Polyak, Benjamin Bolte, Tu-Anh Nguyen, Jade Copet, Alexei Baevski, Abdelrahman Mohamed, and Emmanuel Dupoux. 2021. On generative spoken language modeling from raw audio. *Transactions of the Association for Computational Linguistics*, 9:1336–1354.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*.

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.

Shaoshi Ling, Yuxuan Hu, Shuangbei Qian, Guoli Ye, Yao Qian, Yifan Gong, Ed Lin, and Michael Zeng. 2023. Adapting large language model with speech for fully formatted end-to-end speech recognition.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2023a. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Jiasen Lu, Christopher Clark, Rowan Zellers, Roozbeh Mottaghi, and Aniruddha Kembhavi. 2022a. Unified-io: A unified model for vision, language, and multi-modal tasks. *arXiv preprint arXiv:2206.08916*.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022b. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210.

Zhiliang Peng, Li Dong, Hangbo Bao, Qixiang Ye, and Furu Wei. 2022. Beit v2: Masked image modeling with vector-quantized visual tokenizers. *arXiv preprint arXiv:2208.06366*.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International Conference on Machine Learning*, pages 28492–28518. PMLR.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.

Paul K Rubenstein, Chulayuth Asawaroengchai, Duc Dung Nguyen, Ankur Bapna, Zalán Borsos, Félix de Chaumont Quitry, Peter Chen, Dalia El Badawy, Wei Han, Eugene Kharitonov, et al. 2023. Audiopalm: A large language model that can speak and listen. *arXiv preprint arXiv:2306.12925*.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. Hugginggpt: Solving ai tasks with chatgpt and its friends in huggingface. *arXiv preprint arXiv:2303.17580*.

Mustafa Shukor, Corentin Dancette, Alexandre Rame, and Matthieu Cord. 2023. Unified model for image, video, audio and language tasks. *arXiv preprint arXiv:2307.16184*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

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

Changhan Wang, Anne Wu, and Juan Pino. 2020. Covost 2 and massively multilingual speech-to-text translation. *arXiv preprint arXiv:2007.10310*.

Jian Wu, Yashesh Gaur, Zhuo Chen, Long Zhou, Yimeng Zhu, Tianrui Wang, Jinyu Li, Shujie Liu, Bo Ren, Linquan Liu, and Yu Wu. 2023a. On decoder-only architecture for speech-to-text and large language model integration.

Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. 2023b. Next-gpt: Any-to-any multimodal llm. *arXiv preprint arXiv:2309.05519*.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. 2023. A survey on multimodal large language models. *arXiv preprint arXiv:2306.13549*.

Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. 2021. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*.

Lijun Yu, Yong Cheng, Zhiruo Wang, Vivek Kumar, Wolfgang Macherey, Yanping Huang, David A Ross, Irfan Essa, Yonatan Bisk, Ming-Hsuan Yang, et al. 2023a. Spae: Semantic pyramid autoencoder for multimodal generation with frozen llms. *arXiv preprint arXiv:2306.17842*.

Lijun Yu, José Lezama, Nitesh B Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, et al. 2023b. Language model beats diffusion–tokenizer is key to visual generation. *arXiv preprint arXiv:2310.05737*.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. 2021. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2023a. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. *arXiv preprint arXiv:2305.11000*.

Hang Zhang, Xin Li, and Lidong Bing. 2023b. Video-llama: An instruction-tuned audio-visual language model for video understanding.

Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. 2023c. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*.

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

Yu Zhang, Wei Han, James Qin, Yongqiang Wang, Ankur Bapna, Zhehuai Chen, Nanxin Chen, Bo Li, Vera Axelrod, Gary Wang, Zhong Meng, Ke Hu, Andrew Rosenberg, Rohit Prabhavalkar, Daniel S. Park, Parisa Haghani, Jason Riesa, Ginger Perng, Hagen Soltau, Trevor Strohman, Bhuvana Ramabhadran, Tara Sainath, Pedro Moreno, Chung-Cheng Chiu, Johan Schalkwyk, Françoise Beaufays, and Yonghui Wu. 2023e. Google usm: Scaling automatic speech recognition beyond 100 languages.

Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. 2023f. Multimodal chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*.

Kaizhi Zheng, Xuehai He, and Xin Eric Wang. 2023. Minigpt-5: Interleaved vision-and-language generation via generative vokens. *arXiv preprint arXiv:2310.02239*.

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

A Prompts for different tasks

We present the prompts we use for different tasks in Table 9, which are generated by GPT4 automatically.

B Case study of Coco-Caption

We present several cases that are randomly selected from the development set of Coco-caption. The results are shown in Figure 3.

C K-means Cluster analysis

Table 10 shows the difference when adopting different audio vocab sizes. All the tokenizers are trained based on the features of the 11th layer of $Whisper_{small}$. We find out that the vocab size has a substantial effect on performance. Compared to clustering 1024 tokens, clustering 8192 tokens can result in a WER improvement of over 18 percentage points. This makes the clustering-based discretization approaches more versatile than the VQ-based neural codecs for the audio. The former can adjust the vocabulary size by tuning the number of clustering centers, while the latter needs to retrain a vector quantization module.

Task	Prompts
image caption	Please provide a caption for the image that has been given. Your task is to write a caption for the provided image. The objective is to come up with a caption for the image that has been provided. You are required to write a caption for the provided image. Your job is to create a caption for the image that has been given. The challenge is to think of a caption for the provided image. You have been given an image and your goal is to write a caption for it. You have been given an image and your task is to write a caption for it. The task at hand is to provide a caption for the image that has been provided. Your assignment is to come up with a caption for the provided image.
ASR	Write a response that appropriately completes the request based on the provided audio.
image generation	Create an image that perfectly matches the input sentence. Generate an image that fits the input sentence perfectly. Produce an image that seamlessly complements the input sentence. Create a picture that perfectly corresponds to the input sentence. Generate an image that perfectly aligns with the input sentence. Create an image that perfectly harmonizes with the input sentence. Produce an image that perfectly integrates with the input sentence. Generate an image that perfectly suits the input sentence. Create an image that perfectly matches the input sentence in every way. Produce an image that perfectly corresponds to the input sentence in every aspect.

Table 9: The prompts generated by GPT4 for different tasks.

Vocab Size	1024	2048	4096	8192
WER	40.22	30.85	25.31	**21.49**

Table 10: We randomly sample 500 audio-text pairs from the development set of the CoVoST 2, and the performance with different vocab sizes is shown in the table.

Image id: COCO_val2014_000000200959.jpg

Prompt: The task at hand is to provide a caption for the image that has been provided.

Output: A man with a black jacket flying through a snow-covered slope while riding a snowboard.

Reference: Person on snowboard jumping in air with mountains in the background.

Image id: COCO_val2014_000000384213.jpg

Prompt: You have been given an image and your task is to write a caption for it.

Output: A very small kitchen with a sink, two windows with curtains

Reference: A kitchen is shown with a variety of items on the counters.

Image id: COCO_val2014_000000466052.jpg

Prompt: Please provide a caption for the image that has been given.

Output: A coffee mug sits in the corner on a counter with several tooth brushes and pastes in it.

Reference: A coffee cup filled with tooth paste and toothbrushes.

Figure 3: Some examples in the coco2014 validation set. For each case, we present the original image ID, the prompt, the output of our model, and one reference caption randomly selected among all five references.