Peacock: A Family of Arabic Multimodal Large Language Models and Benchmarks

Fakhraddin Alwajih **El Moatez Billah Nagoudi** **Gagan Bhatia**
Abdelrahman Mohamed **Muhammad Abdul-Mageed**
The University of British Columbia & Invertible AI
{muhammad.mageed@}ubc.ca;invertible.ai

Abstract

Multimodal large language models (MLLMs) have proven effective in a wide range of tasks that require complex reasoning and linguistic comprehension. However, due to a lack of high-quality multimodal resources in languages other than English, success of MLLMs remains relatively limited to English-based settings. This poses significant challenges in developing comparable models for other languages, even those with large speaker populations, such as Arabic. To alleviate this challenge, we introduce a comprehensive family of Arabic MLLMs, dubbed *Peacock*, with strong vision and language capabilities. Through comprehensive qualitative and quantitative analysis, we demonstrate the solid performance of our models on various visual reasoning tasks and further show their emerging dialectal potential. Additionally, we introduce *Henna*, a new benchmark specifically designed for assessing MLLMs on aspects related to Arabic culture, setting the first stone for culturally-aware Arabic MLLMs. The GitHub repository for the *Peacock* project is available at https://github.com/UBC-NLP/peacock.

1 Introduction

Empowered by progress in large language models (LLMs) and foundation models of other modalities, multimodal large language models (MLLMs) now have a remarkable understanding (Alayrac et al., 2022; Li et al., 2023e; Dai et al., 2023; Liu et al., 2023b,a; Zhu et al., 2023). For example, they can handle various complex reasoning tasks spanning from visual question answering to understanding sarcastic comics (Achiam et al., 2023; Yang et al., 2023). These capabilities, however, are mostly seen in models serving the English language. This leaves behind the majority of the world's languages, furthering an already acute technological divide. We alleviate this challenge for Arabic, a diverse collection of languages and dialects with a native population of more than 400 million speakers.

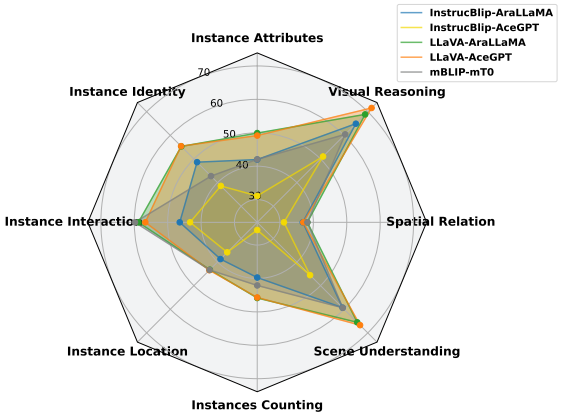

Figure 1: Comparison between the performance of Peacock and mBlip models on SEED-Benchmark dimensions.

More concretely, we draw inspiration from English counterparts (Dai et al., 2023; Liu et al., 2023b) to present a robust family of Arabic MLLMs with powerful vision and language capabilities. Our models adopt the approach of integrating an image encoder with an Arabic text decoder. In our experimental setup, we explore two popular directions for aligning the vision and the language components: one involves employing a fully connected layer as a projection head on top of the vision encoder (Liu et al., 2023b), while the other utilizes a Q-former transformer (Dai et al., 2023). All models are trained in two stages, a pre-training stage and an instruction fine-tuning stage. For the first stage, we curate high-quality pretraining data from publicly available English datasets. We translate these datasets into Arabic and apply a carefully designed pipeline to ensure the quality of our training data. Similarly, we curate and translate an instruction fine-tuning dataset which is essential for achieving reasoning and conversational capabilities.

We showcase the performance of our models across different tasks such as visual question answering (VQA) and visual reasoning. Our models perform much better than a multilingual baseline mBlip (Geigle et al., 2023) on different tasks and datasets, and we set the first comprehensive Arabic vision-language benchmark to facilitate future work in this area. Finally, we demonstrate the promising capabilities of our Peacock models in interacting in dialectal Arabic by conducting a case study on the Egyptian dialect. When fine-tuned on a small set of Egyptian dialect data, our models exhibit an interesting level of proficiency in their dialectal responses when prompted in the same dialect. We hope this acts as a spark for future works in dialectal Arabic vision language models.

To summarize, our contributions in this paper are as follows: **(1)** We introduce a suite of Arabic MLLMs, dubbed *Peacock*, capable of instruction following and visual reasoning, in addition to their intriguing dialectal affinity. For developing Peacock, we use existing vision and language models. We also offer a new language model, *AraLLaMA*, based on LLaMA2-7B (Touvron et al., 2023). **(2)** We introduce a diverse collection of Arabic translated datasets carefully curated for the training and evaluation of Arabic MLLMs. **(3)** We adapt the popular LLaVA (Liu et al., 2023b) benchmark and SEED-Bench (Li et al., 2023d) for Arabic MLLMs evaluation. **(4)** We present *Henna*, a benchmark designed to measure model capabilities in interpreting images related to Arabic culture.

The rest of this paper is organized as follows: In Section 2, we provide an overview of related work. Section 3 introduces Peacock, our family of MLLMs. In Section 4, we describe our evaluation strategies and benchmarks. In Section 5, we present our experiments, human evaluation, and a comprehensive analysis of our models. We conclude in Section 6.

2 Related Work

2.1 Multimodal Large Language Models

Progress in MLLMs is largely dependent on advances in LLMs. Refer to Appendix **??** for more details on LLM-related works. The common trend in recent MLLMs involves integrating an LLM as their text decoder alongside a vision encoder for image understanding. Several approaches were proposed for aligning the vision encoder with the text decoder. Flamingo (Alayrac et al., 2022) and Otter (Li et al., 2023c), for example, blend a **vision encoder with a resampler and a cross-gated attention layer**, reducing the computational load in vision-text cross-attention, and enhancing instruction optimization. While BLIP-2 (Li et al., 2023e) and InstructBLIP (Dai et al., 2023), combine a **vision encoder with a Q-former and a linear layer**, streamlining the cross-modality projection and utilizing learnable query vectors for feature extraction. LLaVA (Liu et al., 2023b,a), on the other hand, pairs a **vision encoder with multilayer perceptron (MLP)**, retaining all visual tokens for comprehensive visual information processing. Finally, the simplest form, illustrated by models such as Fuyu (Bavishi et al., 2023) and OtterHD (Li et al., 2023a), relies **solely on a linear layer**, operating as basic decoder-only transformers without specialized vision encoders. This diversity in design showcases the innovative approaches in integrating vision and language in MLLMs.

2.2 Visual Instruction Tuning

Following the success of instruction tuning in LLMs, recent works in MLLMs transitioned to visual instruction tuning. MULTIINSTRUCT (Xu et al., 2022) pioneered this transition by creating a multi-modal instruction tuning benchmark dataset that transforms 62 different multi-modal tasks into a unified sequence-to-sequence format. Building on this, LLaVA (Liu et al., 2023b) leveraged GPT-4's adeptness in understanding multimodal textual representations to reformulate image-text pairs into an instruction-following format. Similarly, MIMIC-IT (Li et al., 2023b) focused on generating instruction-response pairs using multi-modal in-context information and a variety of visual scenes. Most recently, M3IT (Li et al., 2023f) converted traditional vision-language tasks into a unified vision-to-text framework through manual instruction writing and dataset pre-processing. This includes tasks such as captioning, visual question answering, visual conditioned generation, reasoning, and classification. In their comprehensive survey, Yin et al. (2023) provide an extensive overview of MLLMs, including an evaluation of their performance and capabilities. This paper serves as a valuable resource for researchers interested in the field of MLLMs.

2.3 Arabic MLLMs

The majority of research in Arabic MLLMs focuses mainly on image captioning (ElJundi et al., 2020; Attai and Elnagar, 2020; Afyouni et al., 2021;

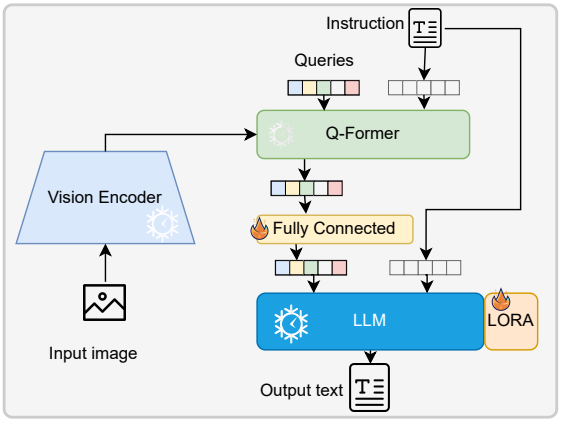

Figure 2: Peacock InstructBLIP architecture: Integrates instruction-specific visual features using Q-Former and a frozen pretrained image encoder.

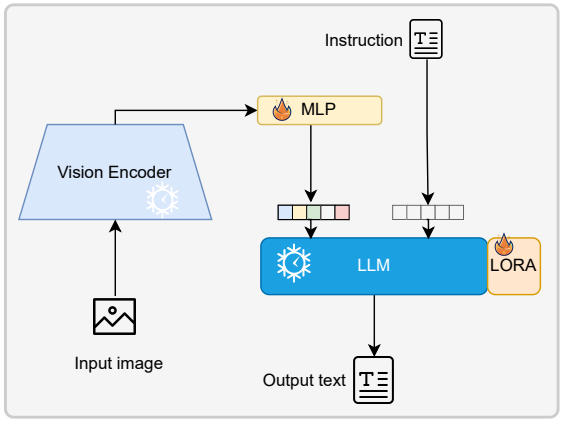

Figure 3: Peacock LLaVA architecture: Combines a pretrained frozen vision encoder with trained Arabic LLMs via an MLP bridge.

Emami et al., 2022; Eddin Za'ter and Talafha, 2022; Elbedwehy and Medhat, 2023; Mohamed et al., 2023). Other areas, VQA for example, remain largely unexplored. This is primarily due to scarcity of publicly available datasets in these areas. As far as we know, the only significant work in Arabic VQA is by Kamel et al. (2023) and explores closed-form VQA without attempting generative VQA. We also know of no native Arabic datasets for either image captioning or VQA, with two exceptions: AraCOCO (Mohamed et al., 2023) for image captioning, which is mainly used for evaluation, and AVQA (Kamel et al., 2023) for VQA, which was automatically generated from MSCOCO for Arabic VQA. In many works, translations of either MSCOCO or Flickr8k are utilized (ElJundi et al., 2020; Afyouni et al., 2021; Mohamed et al., 2023).

3 Peacock

3.1 Architectures

The Peacock family is designed based on the vision components of two architectures, that of InstructBlip (Dai et al., 2023) and LLaVA1.5 (Liu et al., 2023a). For language, our models are integrated with one of two powerful Arabic LLMs, AceGPT (Huang et al., 2023)[1] and a new model based on LLaMA2-7B, dubbed *AraLLaMA*, that we further pretrain on a large Arabic dataset and fine-tune using diverse instructions. Our motivation behind introducing AraLLaMA is to create a model with strong knowledge of the Arabic language and culture. More information about AraLLaMA and how it compares to AceGPT is in Appendix **??**.

InstructBlip-Based Peacock. Here, our models consist of four key components: (1) A vision encoder based on the ViT (ViT/G-14) model (Dosovitskiy et al., 2020), operating at a resolution of 224×224 and employing a patch size of 14. (2) A Querying Transformer (Q-former) (Li et al., 2023e), designed to link the pretrained vision encoder with the LLM, using the BERT base model (Devlin et al., 2018) as its foundation. (3) A linear layer projector, tasked with aligning the output of the Q-former with the LLM embedding space. (4) An LLM, incorporating one of the two forenamed models, AceGPT or AraLLaMA, both of which are derivatives of the LLaMA2 architecture enhanced for Arabic. Figure 2 illustrates this architecture.

LLaVA-Based Peacock. For this setting, models are structured around three primary components: (1) A vision encoder employing the CLIP-Large model (Radford et al., 2021), capable of processing images at a resolution of 336x336 and a patch size of 14, converting these images into 576 tokens. (2) A two-layer MLP projector that aligns the output of the visual and language modalities. (3) And either AceGPT or AraLLaMA. The architecture is shown in Figure 3

3.2 Pretraining

Our models are trained in two stages, a pretraining stage and an instruction fine-tuning stage. The pretraining stage aims to train the alignment module, which projects the visual and textual features into a common embedding space. The models are trained

[1] In all our experiments, we use the *AceGPT-7B-chat*. We also limit ourselves to LLMs with 7B parameters due to computational constraints.

using our carefully curated text-image pairs dataset. In the case of InstructBlip-based models, only the projection layer, which is the alignment module, is trainable. In contrast, the Q-former, vision encoder and language model parameters are frozen. Meanwhile, for the LLaVA-based models, only the MLP connector is the trainable part, with the CLIP encoder and LLM being frozen.

3.3 Visual Instruction Fine-tuning

After the pretraining stage, the model will only be capable of generating simple captions and descriptions of an image. To give the models the ability to function on tasks requiring visual reasoning and engage in an intelligible visual conversation, we further fine-tune them using instruction datasets. To keep computational costs manageable, we employ the parameter-efficient fine-tuning technique LoRA (Hu et al., 2021). Similar to the previous stage, in addition to the LoRA parameters, only the linear layer is trainable in the case of InstructBlip models, while for LLaVA models, we fine-tune the MLP and apply LoRA to the LLM, following the LLaVA 1.5 training scheme (Liu et al., 2023a). We provide in Table **??** the number of parameters for the main components of each model in Appendix **??**.

4 Datasets and Benchmarks

4.1 Translation and Filtering Pipeline

A significant challenge for Arabic MLLMs is lack of available resources, which is due to the difficulty of retrieving relevant Arabic image-text pairs from the internet at scale and absence of suitable image-text relevance filtering methods similar to that of CLIP (Radford et al., 2021)[2]. To address this resource gap, we introduce a careful translate-and-filter pipeline for converting publicly available image-text datasets into Arabic without losing data quality. To this end, we adopt the latest version of Google translate API (Google Cloud). We follow Mohamed et al. (2023) in further assuring high quality of acquired translations using a multilingual sentence embedding model LaBSE (Feng et al., 2020). We calculate the similarity of embeddings between the original and translated sentences (questions and answers), retaining translations that meet a minimum similarity threshold of 80% or greater for both the question and the answer. Figure 4 demonstrates the filtering pipeline. We provide details about our datasets and translation method in Appendix **??**, and sample translations illustrating variations in quality ranging from good to moderate to poor in Figure **??** (also in Appendix **??**).

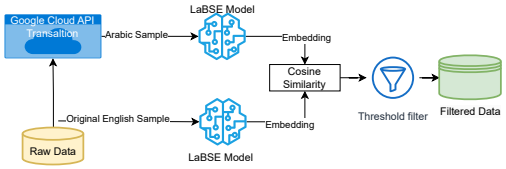

Figure 4: Our data filtering pipeline. After translating the data through Google Cloud API, we obtain the embeddings of both the original and translated samples using the multilingual sentence embedding model LaBSE. For each sample, we calculate the cosine similarity between the two extracted embeddings and reject samples under an 80% threshold.

4.2 Pretraining Data

Aligning with recent work showing that the quality of LLMs pretraining data is more important than quantity (Gunasekar et al., 2023; Lee et al., 2023), we curate a high-quality text-image pairs dataset collected from publicly available sources. Specifically, we utilize LCS-558K (Liu et al., 2023b) and COCO (Lin et al., 2014) as our pretraining data. LCS-558K encompasses 558k text-image pairs carefully curated from three datasets: LAION (Schuhmann et al., 2021), Conceptual Captions (Sharma et al., 2018), and SBU (Ordonez et al., 2011). COCO is a high-quality dataset comprising 118k images, covering 80 different objects, with five captions per image, all human-annotated. As stated in Section 4.1, all the datasets are translated into Arabic using Google API and further filtered based on their semantic similarity with the original English text.

4.3 Instruction Fine-tuning Data

For the second training stage, we curate another dataset that follows the instructions tuning paradigm as in Liu et al. (2023b). Concretely, the model is asked to respond to a specific instruction or question for each image in the dataset. The first dataset we include is the multi-modal instructions dataset by Liu et al. (2023b). It comprises 150k samples covering conversations, detailed image descriptions, and complex reasoning instructions and responses. This dataset was created us-

[2]CLIP was used in filtering many English web scraped large-scale datasets (Ordonez et al., 2011; Sharma et al., 2018; Changpinyo et al., 2021).

ing GPT-4 (Achiam et al., 2023), and the images were taken from the COCO dataset. Additionally, we incorporate the VQAv2 dataset (Goyal et al., 2017) after transforming it to the same instructions and responses format. To incorporate further diverse instructions, we utilize 60k multi-choice questions extracted from LLaVA1.5 mixed instruction dataset (Liu et al., 2023a). This exposes the model to different scenarios, giving it better generalization capabilities. Similar to the pretraining stage, all the datasets are translated using Google API and filtered following our data-cleaning pipeline.

4.4 Evaluation Benchmarks

SEED-Bench. SEED-Bench (Li et al., 2023d) consists of 19K multiple-choice questions, each meticulously annotated by humans. These questions cover 12 evaluation dimensions, addressing the comprehension of both image and video modalities. This study exclusively focuses on the image-only subset of SEED-Bench comprising 14K multiple-choice questions. SEED-Bench is translated via our translation and filtering pipeline as described in Section 4.1.

LLaVA-Bench. The LLaVA-Bench (Liu et al., 2023b) comprises 30 images, which the authors randomly select from the COCO-Val-2014 dataset. For each image, three questions are generated, resulting in 90 instances. These questions fall into three categories: conversational, detailed description, and complex reasoning. This benchmark evaluates the model's performance across conversation, description, and reasoning, using GPT-4 scoring.

Henna Benchmark. As Arabic culture may be underrepresented in current English MLLMs datasets, we develop Henna, a new benchmark for testing purposes only. Henna comprises attractions, food, events, and other Arabic-relevant objects, consisting of 1,132 samples that have been manually curated and reviewed to ensure quality and relevance. More details about how we create Henna and how we use it for evaluation are in Section 5.2.4.

5 Experiments

5.1 Implementation Details

In the **first training stage**, we use the 916k image-text pairs described in Section 4.2 to train Peacock models. The pretraining phase spans three epochs with a batch size of 32 and a learning rate of 1e-3. As previously described, all model parameters are kept frozen except for the projection layer in the case of InstructBlip-based models and the MLP connector for LLaVA-based models. During the **second training phase**, we utilize the instructions dataset introduced in Section 4.3. The models are further fine-tuned for three epochs with a batch size of eight and a learning rate of 2e-5. As mentioned before, only the introduced LoRA parameters are trainable, with the addition of the projection layer in the case of InstructBlip-based models and MLP connector for LLaVA-based models.

For the training objective, we follow the language modeling approach where the model predicts the next text token given previously predicted text tokens and the visual features. Concretely, our goal is to maximize the probability of the next token or, for mathematical convenience, minimize the negative log-likelihood. The loss is calculated only on the response generated by the model. The instructions and visual tokens are skipped during these calculations.

5.2 Results and Discussion

We evaluate our suite of models on a range of typical vision-language tasks and benchmarks. In addition, we show our models' performance on our novel Arabic cultural dataset, Henna. Since this is the first work on Arabic MLLMs, we adapt popular benchmarks in the literature to our case. Mainly, these are a VQA-tasks benchmark, LLaVA-Bench (Liu et al., 2023b), and SEED-Bench (Li et al., 2023d). We also evaluate the performance on Henna benchmark and conduct a case study focusing on the Egyptian dialect. This establishes the first comprehensive benchmark for future works in Arabic MLLMs. We further compare our models with the multilingual mBlip model (Geigle et al., 2023) as a baseline for completeness. The mBlip model is trained on 96 languages, including Arabic.

5.2.1 General VQA

In general VQA tasks, the challenge involves answering textual questions about images, requiring learning and integrating visual and textual information. This demonstrates a deep understanding of the interconnectedness between the two modalities. To evaluate performance in general VQA, our validation process includes three datasets: VQAv2 (Goyal et al., 2017), OKVQA (Marino et al., 2019), and GQA (Hudson and Manning, 2019). Notably, evaluation of English VQA tasks is typically performed through online platforms by submitting results. However, this option is un-

Model	Architecture	LLM	VQAv2		OKVQA		GQA	
			All	Filtered	All	Filtered	All	Filtered
Baseline	**mBlip**	mT0-XL-5B	38.55	50.8	8.59	18.18	35.95	50.45
		BLOOMZ-7B	41.00	55.7	11.87	23.30	38.55	54.85
	**InstructBlip**	AraLLaMA	**44.55**	56.15	**20.97**	**29.77**	**42.60**	**58.05**
		AceGPT	39.00	51.20	10.69	16.82	37.00	57.60
	LLaVA	AraLLaMA	40.85	52.45	14.79	25.57	33.45	49.75
		AceGPT	41.45	**56.65**	15.14	26.36	33.27	52.20

Table 1: The zero-shot performance of our Peacock models against mBlip on the dev set of different VQA datasets. Models are evaluated on the exact match with the open-generation metric, where an answer is considered correct if it matches any ground truth answers. The baseline is mBlip with different LLMs (mT0-XL-5B and BLOOMZ-7B).

available for Arabic-translated data because these platforms only support English and the original datasets. Since the test sets do not contain ground-truth labels, we evaluate held-out validation sets. We follow Geigle et al. (2023) in using "exact match accuracy with open generation" to evaluate our models' output. The metric considers an answer correct if it matches any of the ground-truth answers. Table 1 shows model accuracy on these datasets under the zero-shot setting for both the filtered and unfiltered(All) versions of the datasets.

It is evident from Table 1 that the top-performing Peacock model, InstructBlip with AraLLaMA LLM, significantly outperforms the best version of mBlip, which integrates the BLOOMZ-7B LLM, by an average margin of 4.5 points. A comparative analysis of all models also reveals significant performance improvements when only the filtered high-quality data is included. This enhancement is consistently observed across all models and tasks, highlighting the crucial role of data quality in the effectiveness of these models.

Furthermore, we observe that the choice of the underlying LLM is has a significant impact on performance. This is the case if we compare integrating AraLLaMA to AceGPT in our overall MLLMs. Specifically, the InstructBlip model integrated with AraLLaMA demonstrates superior performance across all tasks and datasets when using either filtered or unfiltered(All) data in training. This performance disparity is likely attributable to the inherent differences in how these LLMs handle Arabic, with AraLLaMA being more effective due to its extensive training and its ability to align with visual information. In addition, it is worth noting that the performance of Peacock models varies considerably depending on the task, with a general trend of models performing better on the VQAv2 task than on OKVQA and GQA. Such variations can be attributed to each task's inherent complexities and specific requirements, including the sophistication of the presented questions and the nature of the required visual understanding.

Architecture	LLM	Conv	DD	CR	Avg
mBlip	BLOOMZ-7B	55.26	47.89	55.43	52.90
InstructBlip	AraLLaMA	**84.56**	**80.00**	**82.11**	**82.27**
	AceGPT	73.28	61.40	72.67	69.13
LLaVA	AraLLaMA	75.62	65.01	72.33	71.07
	AceGPT	77.81	68.85	73.89	73.61

Table 2: Performance of Peacock models and mBlip on LLaVA-Bench scored by GPT-4. Conv: Conversation. DD: Details Description. CR: Complex Reasoning.

5.2.2 LLaVA-Bench

For evaluation using LLaVA-Bench, we follow the method of Liu et al. (2023b). Table 2 displays our models' successful performance across the three metrics of the LLaVA-Bench. Despite the limited data and resources, this suggests a burgeoning capability for multi-modal comprehension in Arabic. Under the same training conditions, the integration of InstructBlip with AraLLaMA notably excels within the Peacock suite. It achieves an average score of 82.27 on the GPT-4 scale, a significant 9.4 margin over the LLaVA model combined with AceGPT. As shown in Table 2, all Peacock models surpass the mBlip-BLOOMZ-7B baseline in the three metrics of the LLaVA-Bench.

5.2.3 SEED-Bench

For our third benchmark, we adapt the SEED-Bench for Arabic and use it to evaluate our models. Table 3 and Figure 1 present an evaluation of Peacock models across a broad spectrum of visual understanding dimensions within SEED-Bench,

Architecture	LLM	IA	II	IN	IL	IC	SU	SR	VR
mBlip	mT0-XL-5B	42.04	42.76	**59.79**	43.35	42.09	59.37	**38.20**	60.42
InstructBlip	AraLLaMA	**49.91**	55.33	**58.76**	43.25	**45.85**	65.52	**38.20**	68.88
	AceGPT	49.16	**55.43**	56.7	**43.46**	45.69	**66.72**	36.99	**71.60**
LLaVA	AraLLaMA	41.98	48.66	46.39	38.75	39.72	59.34	36.83	64.95
	AceGPT	31.10	38.61	43.30	35.89	25.50	45.57	31.20	51.06

Table 3: Evaluation of mBlip and Peacock models on SEED-Bench across various attributes: Instance Attributes (IA), Instance Identity (II), Instance Interaction (IN), Instance Location (IL), Instances Counting (IC), Scene Understanding (SU), Spatial Relation (SR), and Visual Reasoning (VR).

where a diverse range of performance efficiencies is observed. LLaVA-AraLLaMA emerges as a particularly robust model, excelling in visual reasoning and scene understanding with accuracy scores of 68.88% and 65.52%, respectively. However, it displays weaknesses in spatial relations and instance location. Mirroring this trend, LLaVA-AceGPT showcases strengths in scene understanding and visual reasoning (66.72% and 71.6%, respectively), but marginally underperforms in instance interaction and spatial relations compared to LLaVA-AraLLaMA. In contrast, InstructBlip-AraLLaMA, while proficient in scene understanding and Visual Reasoning (59.34% and 64.95%), falls short in Instance Attributes and Counting, resulting in a lower overall accuracy of 46.43%. InstructBlip-AceGPT, the model with the most modest performance, achieves its best results in visual reasoning and instance interaction (51.06% and 43.3%), but struggles significantly with instance counting and scene understanding. In contrast to mBlip, which outperforms Peacock models only in one dimension (instance interaction) and achieves the same score in the spatial relation dimension as InstructBlip-AraLLaMA.

This comparative analysis underscores the superiority of LLaVA-based models in the Peacock family on SEED-Bench, especially those with AraLLaMA, over InstructBlip models in most tasks. This could be attributed to the capability of AraLLaMA in understanding and ability of align information coming from the vision encoder on the one hand and the input questions about the input image, on the other hand. Meanwhile, the InstructBlip models, particularly those with AceGPT LLM, reveal limitations in broader visual understanding tasks. The marked variation in performance between AraLLaMA and AceGPT within the same model base highlights the significant impact of language model selection on visual task performance,

Architecture	LLM	Helpfulness	Relevance	Accuracy	Level of Details
mBlip	mT0-XL-5B	34.11	39.15	35.11	20.74
InstructBlip	AraLLaMA	**62.34**	**68.97**	**49.68**	**49.83**

Table 4: Evaluation of InstructBlip-AraLLaMA against mBlip-mt0-xl models on ***Henna***, using GPT-4.

offering valuable insights into the inherent abilities (and limitations) in contemporary MLLMs.

5.2.4 Henna Benchmark

Henna was developed to establish a standard for evaluating Arabic MLLMs on elements particularly related to Arabic culture, such as food, customs, and landmarks. The dataset was created by prompting GPT-4V (OpenAI, 2023) to generate descriptions of images based on questions, while providing it with relevant Wikipedia context. Images were carefully selected to represent the culture of 11 Arab countries.

To achieve this, we selected images from Wikipedia and corresponding articles to create the context for GPT-4V during the generation process. The images represent a range of countries, including Algeria, Egypt, Iraq, Jordan, Morocco, Palestine, Saudi Arabia, Syria, Tunisia, the United Arab Emirates, and Yemen. We identified ten top attractions from each country within categories such as traditional food and cuisine, local customs, historical monuments and sites, common activities and lifestyles, and architectural styles and notable buildings. Figure **??** demonstrates selected examples from the dataset's images.

For each attraction, we used GPT-4V to generate ten questions. Each image was accompanied by its Wikipedia article to provide comprehensive context. This approach yielded a minimum of ten images per country, resulting in a total of 1,132 question-answer pairs across all countries. An example of the dataset generation process is illustrated in Figure 5, and Figure **??** demonstrates four randomly selected images with a generated pair of a question and their answers. Moreover, Figure **??** shows examples of questions and answers generated by the Henna pipeline from the image depicted in Figure **??**. These questions and answers were translated into English to provide a qualitative assessment of the dataset's quality.

The evaluation process utilizes GPT-4 to assess each model's responses based on four criteria: *Helpfulness*, *Relevance*, *Accuracy*, and *Level of Details*. Each criterion is rated on a scale from one to ten, with higher scores indicating better responses.

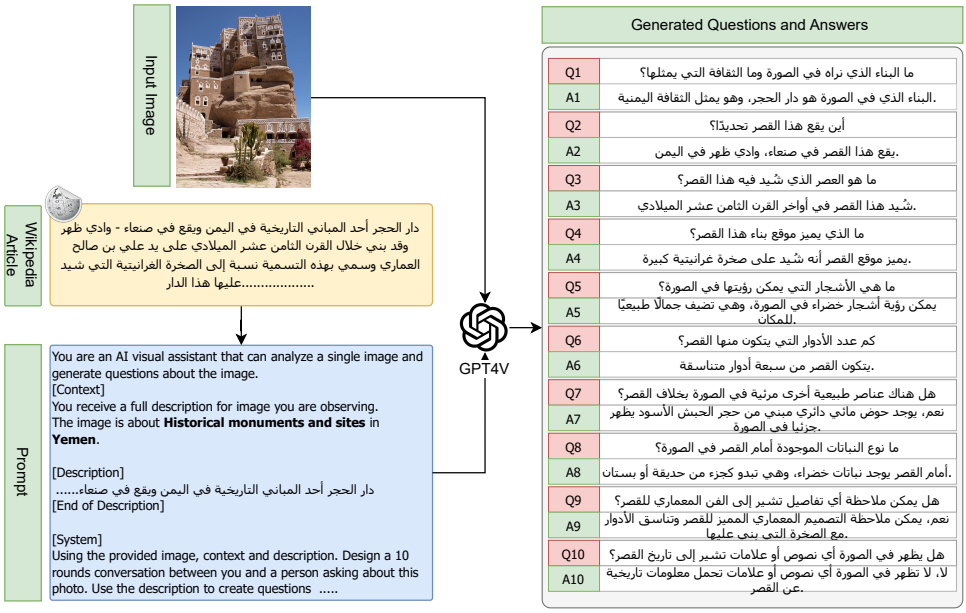

Figure 5: Dataset Generation Example using GPT-4V. This figure demonstrates the process of generating a question-answer dataset for an attraction in Yemen as an example. For each site, an image and its corresponding Wikipedia article were used to provide GPT-4V with rich contextual information. The model then generated ten contextually relevant questions and answers per image.

The evaluation process involves GPT-4 reviewing a question and its correct answer in Arabic, followed by the model's response, which is then rated according to the aforementioned criteria. The results are formatted as a JSON object with keys corresponding to each criterion. Figure **??** illustrates examples where GPT-4's evaluations varied in quality, showing high, medium, and low evaluations for different models' responses. An example of the prompt used in the evaluation is shown in Figure **??**.

The leading model from the Peacock suite was evaluated against the multilingual model mBlip following our benchmark. The data presented in Table 4 demonstrates the superiority of the InstructBlip model paired with AraLLaMA, setting a benchmark for future models in terms of their ability to comprehend and recognize aspects of Arabic culture. Figure 6 shows an example response from Peacock along with a response from GPT-4V, illustrating the practical application of these findings.

This structured evaluation method, where GPT-4 serves both as the subject and the evaluator, facilitates a quantitative analysis of the model's performance in understanding and responding to visual questions in Arabic.

5.2.5 Qualitative Analysis

In our qualitative analysis of Peacock models, we select two random samples from each question type previously described in Section 5.2.2, totaling six samples. Figure **??** displays the answers by all Peacock models to these six questions, accompanied by their corresponding images.

For the conversion type questions, one direct question involves asking about the color of an elephant. While InstructBlip integrated with AceGPT fails to provide the correct answer, all other models succeed. In the second conversion example, the LLaVA-based models are unable to answer a question about counting donuts. In the detail type questions, all models provide answers that are closely related to the details of the objects in the images, albeit with some hallucinations. For the complex type questions, all models provide subjective answers, which, despite offering slightly different conclusions about the image, can still be considered correct. In summary, InstructBlip integrated with AraLLaMA, provides accurate and more helpful answers for most of the three types of questions.

5.2.6 Case Study with Egyptian Dialect

Attention to dialectal Arabic in the NLP research community is not sufficient to date, with complete absence when it comes to MLLMs. Addressing this gap, we conduct the first study on the capabilities of MLLMs in generating dialectal Arabic, focusing the study on the Egyptian dialect. Out of the box, our fine-tuned models were able to under-

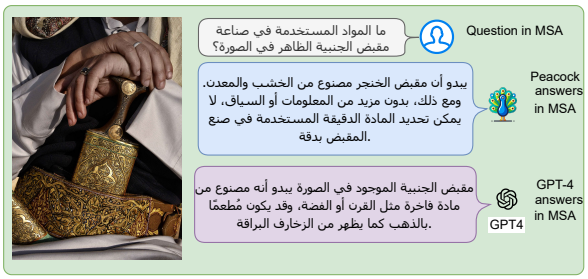

Figure 6: Examples of responses from Peacock and GPT-4V regarding an image related to Yemeni culture.

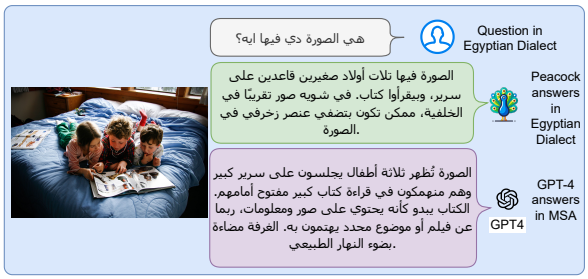

Figure 7: Both Peacock and GPT-4V accurately respond to a question in the Egyptian dialect. While GPT-4V provides a slightly more detailed answer, it does so in MSA. In contrast, Peacock's response is in the same Egyptian dialect as the question.

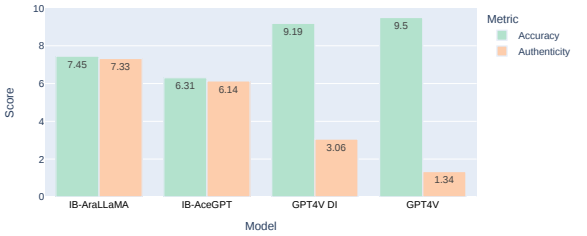

Figure 8: Human evaluation results on the accuracy and authenticity of model responses to questions about images in Egyptian dialect. "IB-AraLLaMA " denotes our InstructBlip-AraLLaMA model, and "IB-AceGPT" refers to our InstructBlip-AceGPT model. "GPT-4V DI" is the GPT-4V model explicitly instructed to respond in the Egyptian dialect."GPT-4V" represents the GPT-4V model, which is given a question in the Egyptian dialect, similar to how Peacock models are instructed.

stand questions posed in the Egyptian dialect but responds in MSA. Following this observation, we transform a subset of 1k random samples from our instruction tuning dataset into Egyptian dialect by a professional Egyptian translator. This small dataset is then used to further fine-tune our InstructBlip based Peacock models following the previously mentioned experimental setup. Surprisingly, as seen in Figure 7, our Peacock models are capable of generating solid answers in Egyptian dialect when instructed on this small sample, while keeping their MSA fluency intact. This could be due to the fact that our LLMs have seen dialectal Egyptian Arabic during their pretraining.

To provide a measurable evaluation, we further transform 20 samples from our instruction tuning evaluation set into Egyptian dialect. Using these samples, we appoint four native Egyptian speakers to anonymously score the responses of our models against GPT-4. The evaluation was based on two criteria: *the accuracy* of the model's response to the question (rated on a scale from 1 to 10) and *the authenticity* of the Egyptian dialect (rated on a scale of 1 to 10).

To ensure transparency, the answers from models were anonymized before being presented to the annotators. As shown in Figure 8, Peacock models exhibited greater closeness to the Egyptian dialect compared to GPT-4V, even when the latter was specifically instructed to respond in the Egyptian dialect. On the other hand, our dialectal models lag slightly in the accuracy of the answers, which we assume can be alleviated by providing sufficient training data, a task we leave for future work. More details and examples on the case study are provided in Appendix **??**.

6 Conclusion

In this work, we present the family of Peacock models. Peacock demonstrates significant advancements in Arabic MLLMs, showcasing remarkable abilities in interpreting visual data in Arabic language. These models bridge the gap in multimodal understanding capabilities for Arabic and Egyptian dialects by introducing a suite of models, with various reasoning skills, accompanied by a diverse collection of datasets and benchmarks carefully prepared. This includes our Henna benchmark, designed to assess MLLM tasks focused on the Arabic culture. The development of Peacock sets strong baselines and a new benchmark for future work in Arabic MLLMs, highlighting the importance of high-quality data processing and the selection of language models for multimodal task performance.

7 Limitations

We identify a number of limitations for our work, as follows:

- Peacock models have demonstrated remarkable abilities in interpreting visual data in Arabic. However, these models can struggle with object hallucination, where the generated descriptions or answers may include references to objects that do not exist in the input image, along with unnecessary details.
- Additionally, translation errors can significantly impact the model's performance and propagate through the training data. We have identified several such errors in the model's responses. For example, the English word 'sitting' typically indicates the location of an object. However, the Google API often mistranslates it to suggest that the object is lying down, as seen in the translation of 'The train is sitting at the station' to القطار جالس في المحطة, where جالس inaccurately implies that the train is lying down.
- Moreover, the Peacock model's capabilities are further limited in recognizing text within images. This limitation stems from the fact that our training datasets do not include image-text pairs.

8 Ethics Statement

Energy Efficiency. Our Peacock models, like many large MLLMs, require significant pretraining time and are not energy-efficient. We acknowledge this critical issue and support continued research towards developing energy-efficient models.

Data. Our pretraining datasets are translated from publicly available English data, encompassing diverse genres, communities and varieties. Our Peacock models demonstrate potential in applications involving several Arabic varieties, serving broad populations.

Human Annotation. Three authors of this paper, all Arabic native speakers with PhD degrees and extensive NLP experience, conducted the human annotation. They are full-time employees of our research group, with data annotation as part of their job duties. No IRB review was necessary as the project used publicly available data without requiring access to any private accounts.

Applications. While Peacock, like many MLLMs, can be misused, it also holds promise for beneficial applications in education, health, and more. Responsible deployment and use are crucial to maximizing its positive impact. It would also help keep Arabic varieties in use in written form in the digital age.

AI Usage. ChatGPT was used to corect grammar in some early stages of the paper writing by some of the authors. This utilization was strictly for the purpose of enhancing the linguistic precision. Our research team independently carried out the fundamental research, analysis, development, and paper writing.

Acknowledgments

We acknowledge support from Canada Research Chairs (CRC), the Natural Sciences and Engineering Research Council of Canada (NSERC; RGPIN-2018-04267), the Social Sciences and Humanities Research Council of Canada (SSHRC; 435-2018-0576; 895-2020-1004; 895-2021-1008), Canadian Foundation for Innovation (CFI; 37771), Digital Research Alliance of Canada,[3] and UBC ARC-Sockeye.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Imad Afyouni, Imtinan Azhar, and Ashraf Elnagar. 2021. Aracap: A hybrid deep learning architecture for arabic image captioning. *Procedia Computer Science*, 189:382–389.

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.

Anfal Attai and Ashraf Elnagar. 2020. A survey on arabic image captioning systems using deep learning models. In *2020 14th International Conference on Innovations in Information Technology (IIT)*, page 114–119. IEEE.

Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Sağnak Taşırlar. 2023. Fuyu-8b: A multimodal architecture for ai agents. https://www.adept.ai/blog/fuyu-8b. Accessed: 2024-01-01.

[3] https://alliancecan.ca

Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. 2021. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3558–3568.

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

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

Muhy Eddin Za'ter and Bashar Talafha. 2022. Benchmarking and improving arabic automatic image captioning through the use of multi-task learning paradigm. *arXiv e-prints*.

Samar Elbedwehy and T Medhat. 2023. Improved arabic image captioning model using feature concatenation with pre-trained word embedding. *Neural Computing and Applications*, page 1–17.

Obeida ElJundi, Mohamad Dhaybi, Kotaiba Mokadam, Hazem M Hajj, and Daniel C Asmar. 2020. Resources and end-to-end neural network models for arabic image captioning. In *VISIGRAPP (5: VISAPP)*, page 233–241.

Jonathan Emami, Pierre Nugues, Ashraf Elnagar, and Imad Afyouni. 2022. Arabic image captioning using pre-training of deep bidirectional transformers. In *Proceedings of the 15th International Conference on Natural Language Generation*, page 40–51.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic bert sentence embedding. *arXiv preprint arXiv:2007.01852*.

Gregor Geigle, Abhay Jain, Radu Timofte, and Goran Glavaš. 2023. mblip: Efficient bootstrapping of multilingual vision-llms. *arXiv preprint arXiv:2307.06930*.

Google Cloud. Google translation api. `https://cloud.google.com/translate`. Accessed: 2023-12-01.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. 2023. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*.

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. *arXiv preprint arXiv:2106.09685*.

Huang Huang, Fei Yu, Jianqing Zhu, Xuening Sun, Hao Cheng, Dingjie Song, Zhihong Chen, Abdulmohsen Alharthi, Bang An, Juncai He, Ziche Liu, Zhiyi Zhang, Junying Chen, Jianquan Li, Benyou Wang, Lian Zhang, Ruoyu Sun, Xiang Wan, Haizhou Li, and Jinchao Xu. 2023. Acegpt, localizing large language models in arabic.

Drew A Hudson and Christopher D Manning. 2019. 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*, pages 6700–6709.

Sarah M Kamel, Shimaa I Hassan, and Lamiaa Elrefaei. 2023. Vaqa: Visual arabic question answering. *Arabian Journal for Science and Engineering*, pages 1–21.

Ariel N Lee, Cole J Hunter, and Nataniel Ruiz. 2023. Platypus: Quick, cheap, and powerful refinement of llms. *arXiv preprint arXiv:2308.07317*.

Bo Li, Peiyuan Zhang, Jingkang Yang, Yuanhan Zhang, Fanyi Pu, and Ziwei Liu. 2023a. Otterhd: A high-resolution multi-modality model. *arXiv preprint arXiv:2311.04219*.

Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Fanyi Pu, Jingkang Yang, Chunyuan Li, and Ziwei Liu. 2023b. Mimic-it: Multi-modal in-context instruction tuning. *arXiv preprint arXiv:2306.05425*.

Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. 2023c. Otter: A multi-modal model with in-context instruction tuning. *arXiv preprint arXiv:2305.03726*.

Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. 2023d. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv preprint arXiv:2307.16125*.

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

Lei Li, Yuwei Yin, Shicheng Li, Liang Chen, Peiyi Wang, Shuhuai Ren, Mukai Li, Yazheng Yang, Jingjing Xu, Xu Sun, Lingpeng Kong, and Qi Liu. 2023f. M^3it: A large-scale dataset towards multi-modal multilingual instruction tuning.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 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*, pages 740–755. Springer.

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

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 3195–3204.

Abdelrahman Mohamed, Fakhraddin Alwajih, El Moatez Billah Nagoudi, Alcides Alcoba Inciarte, and Muhammad Abdul-Mageed. 2023. Violet: A vision-language model for arabic image captioning with gemini decoder. *arXiv preprint arXiv:2311.08844*.

OpenAI. 2023. Gpt-4v(ision) system card.

Vicente Ordonez, Girish Kulkarni, and Tamara Berg. 2011. Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. 2021. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565.

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

Zhiyang Xu, Ying Shen, and Lifu Huang. 2022. Multiinstruct: Improving multi-modal zero-shot learning via instruction tuning. *arXiv preprint arXiv:2212.10773*.

Zhengyuan Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. 2023. The dawn of lmms: Preliminary explorations with gpt-4v (ision). *arXiv preprint arXiv:2309.17421*, 9(1).

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*.

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*.