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

Anonymous ACL submission

Abstract

Multimodal large language models (MLLMs) have proven effective in a wide range of tasks requiring 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, including 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.

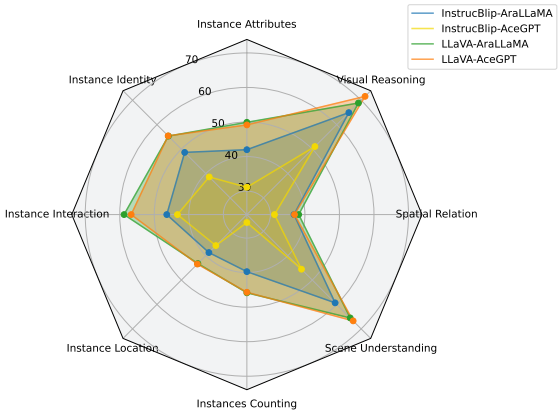

Figure 1: Comparison between the performance of Peacock models on SEED-Benchmark dimensions.

1 Introduction

Empowered by progress in large language models (LLMs) and foundation models of other modalities, multimodal large language models (MLLMs) can now have 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 wide collection of languages and dialects with a native population of more than 400 million speakers.

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 pretraining stage and an instruction finetuning 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 finetuning 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 finetuned 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 MMLLs, 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 A.2 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.), 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 focus mainly on image captioning (ElJundi et al., 2020; Attai and Elnagar, 2020; Afyouni et al., 2021; 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

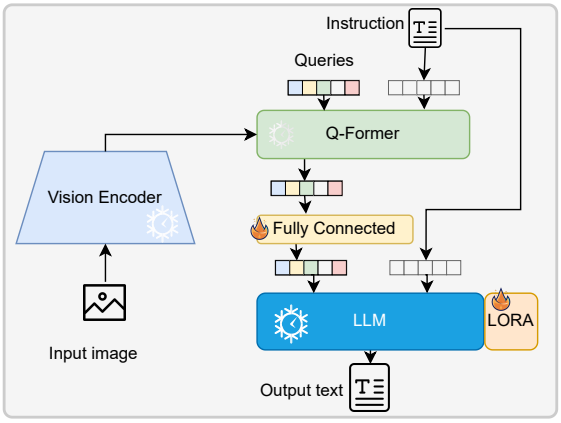

Figure 2: Peacock InstructBLIP architecture: Integrates instruction-specific visual features using Q-Former and a frozen pretrained image encoder.

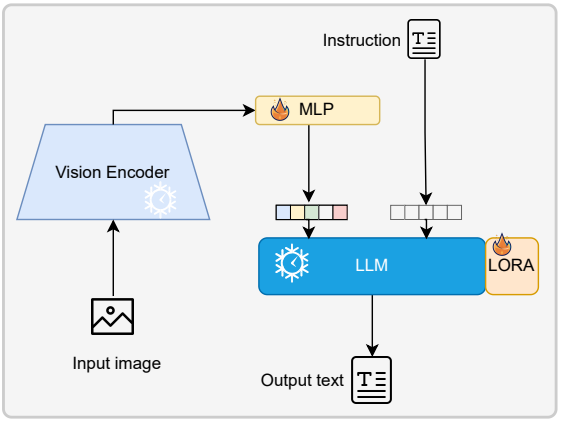

Figure 3: Peacock LLaVA architecture: Combines a pretrained frozen vision encoder with trained Arabic LLMs via an MLP bridge.

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 finetune 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 A.4.

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 finetuning stage. The pretraining stage aims to train the alignment module, which projects the visual and textual features into a common space. The models are trained 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. Mean-

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

while, for the LLaVA-based models, only the MLP is the trainable part, with the CLIP encoder and LLM being frozen.

3.3 Visual Instruction Finetuning

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 finetune them using instruction datasets. To keep computational costs manageable, we employ the parameter-efficient finetuning 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 also freeze the MLP following the LLaVA 1.5 training scheme (Liu et al., 2023a). We provide in Table A.6 the number of parameters for the main components of each model in Appendix A.5.

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 modelLaBSE (Feng et al., 2020). We calculate the similarity of embeddings between the original and translated sentences, keeping translations with a minimum similarity threshold $> 80\%$ to source data. Figure 4 demonstrates the filtering pipeline. Table 1 presents the statistics of the translated data before and after the filtering process. As the table indicates, the reduction ratio due to filtering is approximately 20%. We provide details about our translation method in Appendix A.3, and sample translations illustrating variations in quality ranging from good to moderate to poor in Figure A.8 (also in Appendix A.3).

[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).

Dataset	Type	Non Filtered			Filtered		
		Train	Dev	Test	Train	Dev	Test
COCO2017	Captioning	590k	25k	–	527k	22k	–
LLaVA Pretrain (LCS)	Captioning	558k	–	–	389k	–	–
LLaVA instruct 150k	Instructions	271k	–	–	204k	–	–
VQAv2	VQA	440k	214k	–	351k	172k	–
OKVQA	VQA	9k	4k	–	7k	3k	–
GQA	VQA	938k	132k	–	638k	89k	–
SEED-Bench	Benchmark	–	–	14k	–	–	7k
LLaVA-Bench	Benchmark	–	–	90	–	–	90
Henna-Bench	Benchmark	–	–	–	–	–	1k

Table 1: Breakdown of our publicly released pretraining and instruction finetuning Arabic datasets. Only the filtered datasets highlighted in green and cyan are used for pretraining and instruction finetuning, respectively.

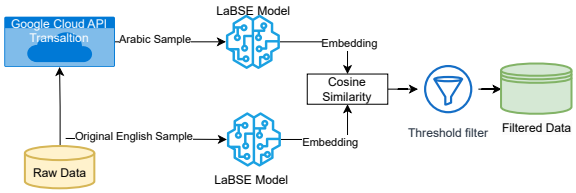

Figure 4: Our data filtering pipeline. After translating the data through Google Cloud API, we obtain the embeddings of both the original and translated samples using the multilingual sentence embedding model LaBSE. For each sample, we calculate the cosine similarity between the two extracted embeddings 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 Finetuning 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 using 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 Benchmark. SEED Benchmark (Li et al., 2023d) (SEED-Bench) 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 Benchmark. The LLaVA benchmark (Liu et al., 2023b) (LLaVA-Bench) 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 Appendix A.6.

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 for LLaVA-based models. During the **second training phase**, we utilize the instructions dataset introduced in Section 4.3. The models are further finetuned 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. 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 benchmark (Liu et al., 2023b), and SEED benchmark (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.

Model	Architecture	LLM	VQAv2		OKVQA		GQA	
			Non Filtered	Filtered	Non Filtered	Filtered	Non Filtered	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
Peacock	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 2: 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).

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 unavailable 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 2 shows model accuracy on these datasets under the zero-shot setting for both the filtered and unfiltered versions of the datasets.

It is evident from Table 2 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 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.

5.2.2 LLaVA Benchmark

For evaluation using LLaVA benchmark, we follow the method of Liu et al. (2023b). Table 3 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 3, all Peacock models surpass the mBlip-BLOOMZ-7B baseline in the three metrics of the LLaVA-Bench.

5.2.3 SEED Benchmark

For our third benchmark, we adapt the SEED-Bench for Arabic and use it to evaluate our models. Table 4 and Figure 1 present an evaluation of Peacock models across a broad spectrum of visual understanding dimensions within SEED-Bench,

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

Architecture	LLM	IA	II	IN	IL	IC	SU	SR	VR
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 4: Evaluation of 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.

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,

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

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

5.2.4 Henna Benchmark

We develop Henna to further evaluate Arabic MLLMs on elements particularly related to the Arabic culture. This includes artifacts such as food, customs, and landmarks. The dataset is created by prompting GPT-4V (OpenAI, 2023) to generate descriptions of images based on questions while providing it (i.e., GPT-4V) with Wikipedia context relevant to the image. The images are carefully picked to represent the culture of a total of 11 Arab countries, including Egypt, Palestine, Morocco, and Yemen. The evaluation process further utilizes GPT-4. For each model response, we prompt GPT-4 to review it 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. More details regarding the images, descriptions, generation process, and evaluation can be found in Appendix A.6.

The leading model from the Peacock suite is evaluated against multilingual model mBlip, following our benchmark. The data presented in Table 5 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 5 shows an example response from Peacock along with a response from GPT-4V, illustrating the practical application of these findings. Detailed examples for this evaluation are further provided in Appendix A.6.

5.2.5 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 capa-

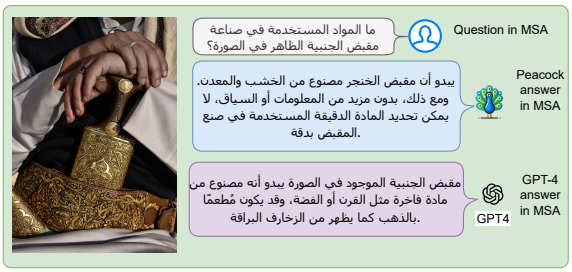

Figure 5: Examples of responses from Peacock and GPT-V4 regarding an image related to Yemeni culture.

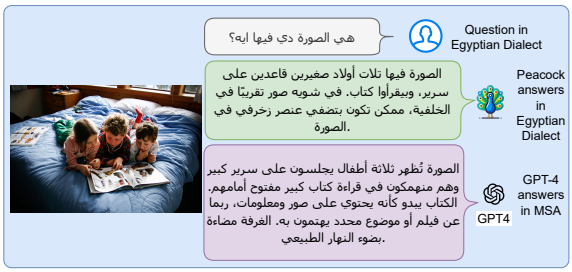

Figure 6: Both Peacock and GPT-4V accurately respond to a question in the Egyptian dialect. While GPT-4V provides a slightly more detailed answer, it does so in MSA. In contrast, Peacock's response is in the same Egyptian dialect as the question.

bilities of MLLMs in generating dialectal Arabic, focusing the study on the Egyptian dialect. Out of the box, our finetuned models were able to understand 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 finetune our InstructBlip based Peacock models following the previously mentioned experimental setup. Surprisingly, as seen in Figure 6, our Peacock models are capable of generating a 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 7, 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 A.8.

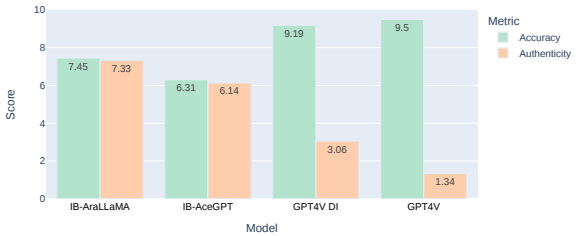

Figure 7: Human evaluation results on the accuracy and authenticity of model responses to questions about images in Egyptian dialect. "IB-AraLLaMA " denotes our InstructBlip-AraLLaMA model, and "IB-AceGPT" refers to our InstructBlip-AceGPT model. "GPT-4V DI" is the GPT-4V model explicitly instructed 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.

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 through the 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 through the Arabic language. 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.
Model Release. We intend to release our models responsibly. We will be releasing a demo for these models later on.
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 spread the Arabic language and preserve some dying language varieties.
AI Usage. It's important to recognize the use of AI tools like ChatGPT in our project. ChatGPT's role was limited and focused primarily on grammar correction in our documentation. This utilization was strictly for the purpose of enhancing the linguistic precision and readability of our texts. We emphasize that our research team carried out the fundamental research, analysis, and development independently.

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.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

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)*, pages 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.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

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.

ChatGPT. 2024. Openai. www.openai.com. Version GPT-4.

Zhihong Chen, Shuo Yan, Juhao Liang, Feng Jiang, Xiangbo Wu, Fei Yu, Guiming Hardy Chen, Junying Chen, Hongbo Zhang, Li Jianquan, et al. Multilingualsift: Multilingual supervised instruction fine-tuning, july 2023b. *URL https://github.com/FreedomIntelligence/MultilingualSIFT. git*.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

W Dai, J Li, D Li, AMH Tiong, J Zhao, W Wang, B Li, P Fung, and S Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. arxiv 2023. *arXiv preprint arXiv:2305.06500*.

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*, pages arXiv–2202.

Ibrahim Abu El-Khair. 2016. 1.5 billion words arabic corpus. *arXiv preprint arXiv:1611.04033*.

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

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Karima Kadaoui, Samar M Magdy, Abdul Waheed, Md Tawkat Islam Khondaker, Ahmed Oumar El-Shangiti, El Moatez Billah Nagoudi, and Muhammad Abdul-Mageed. 2023. Tarjamat: Evaluation of bard and chatgpt on machine translation of ten arabic varieties. *arXiv preprint arXiv:2308.03051*.

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^{3}it: A large-scale dataset towards multi-modal multilingual instruction tuning.

Shenggui Li, Hongxin Liu, Zhengda Bian, Jiarui Fang, Haichen Huang, Yuliang Liu, Boxiang Wang, and Yang You. 2023g. Colossal-ai: A unified deep learning system for large-scale parallel training. In *Proceedings of the 52nd International Conference on Parallel Processing*, pages 766–775.

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*.

El Moatez Billah Nagoudi, Muhammad Abdul-Mageed, AbdelRahim Elmadany, Alcides Alcoba Inciarte, and Md Tawkat Islam Khondaker. 2022. Jasmine: Arabic gpt models for few-shot learning. *arXiv preprint arXiv:2212.10755*.

El Moatez Billah Nagoudi, AbdelRahim Elmadany, Muhammad Abdul-Mageed, Tariq Alhindi, and Hasan Cavusoglu. 2020. Machine generation and detection of arabic manipulated and fake news. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 69–84.

Lorenz Nigst, Maxim Romanov, Sarah Bowen Savant, Masoumeh Seydi, and Peter Verkinderen. 2020. Openiti: A machine-readable corpus of islamicate texts. *nd http://doi. org/10.5281/zenodo*, 4075046.

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.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

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.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

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*.

Neha Sengupta, Sunil Kumar Sahu, Bokang Jia, Satheesh Katipomu, Haonan Li, Fajri Koto, Osama Mohammed Afzal, Samta Kamboj, Onkar Pandit, Rahul Pal, et al. 2023. Jais and jais-chat: Arabic-centric foundation and instruction-tuned open generative large language models. *arXiv preprint arXiv:2308.16149*.

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.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipeline for processing huge corpora on medium to low resource infrastructures. In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*. Leibniz-Institut für Deutsche Sprache.

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

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

Wikipedia contributors. 2024. Wikipedia: The Free Encyclopedia. https://www.wikipedia.org/. [Online; accessed 2024-01-15].

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*.

Imad Zeroual, Dirk Goldhahn, Thomas Eckart, and Abdelhak Lakhouaja. 2019. Osian: Open source international arabic news corpus-preparation and integration into the clarin-infrastructure. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 175–182.

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 Appendices

Appendices

We provide an addition organized as follows:

- Image Attribution A.1.
- Large Language Models A.2.
- Translation and Filtering A.3.
- AraLLaMA A.4.
- Models Parameters A.5.
- Henna Generation pipeline and Examples A.6.
- Qualitative Analysis A.7.
- Case Study with Egyptian Dialect Details and Examples A.8.

A.1 Image Attribution

All figures presented in this document are sourced from the COCO dataset and Wikipedia (Wikipedia contributors, 2024), unless otherwise specified.

A.2 Large Language Models

The field of LLMs has seen significant growth due to increased data availability and computational capabilities. Initial models such as BERT (Devlin et al., 2018) and T5 (Raffel et al., 2020), along with decoder-focused models like GPT (Radford et al., 2019), utilized the Transformer architecture (Vaswani et al., 2017) to achieve remarkable results in various NLP tasks. The advent of GPT3 (Brown et al., 2020) marked a shift towards decoder-only structures, emphasizing auto-regressive decoding for prediction generation. Following models, including PaLM (Anil et al., 2023), expanded the scope of parameters and dataset sizes. Meanwhile, models like InstructGPT (Ouyang et al., 2022), mistral (Jiang et al., 2023), ChatGPT (ChatGPT, 2024) integrated fine-tuning and reinforcement learning to enhance conversational abilities (Ouyang et al., 2022). More recently, Direct Preference Optimization (DPO) technique has been developed to fine-tune language models (LMs) to align with human preferences (Rafailov et al., 2023). The advancements in LLMs were not limited to the English language only, recent works in Arabic LLMs showed promising performance. (Nagoudi et al., 2022) presented Jasmine, a strong Arabic text decoder capable of performing Arabic text generation and classification tasks. While (Huang et al., 2023; Sengupta et al., 2023) followed (ChatGPT, 2024) instructions finetuning and introduced language decoders based on LLaMA2 (Touvron et al., 2023) capable of following instructions and holding a coherent conversation in Arabic. These advancements, coupled with contributions from the open-source community, have established new standards and created new opportunities for research in the NLP field.

A.3 Translation and Filtering

English	Arabic Translation	Similarity
A sidewalk that has a microwave sitting on it	رصيف به ميكروويف	0.305786
A elephant reaching out with some thing with its nose.	فيل يمد يده بشيء بأنفه.	0.493334
A man standing in front of a clock.	رجل يقف أمام الساعة.	0.966210

Figure A.8: Three examples illustrating the variations in translation quality from English to Arabic, ranging from good to moderate to poor, as indicated by the similarity scores.

We also assessed translations by the No Language Left Behind (NLLB) project (Costa-jussà et al., 2022) In the literature, ElJundi et al. (2020) found that Google translates API translations to be sub-optimal, while Mohamed et al. (2023) showed that NLLB performs better than the free Google translate API integrated into Google Sheets. However, our investigation reveals that the latest Google Translate API has superior performance and provides higher-quality translations than NLLB. This observation aligns with the recent findings of Kadaoui et al. (2023).

A.4 AraLLaMA

Pretraining Data. AraLLaMA, is an autoregressive pretrained language model based on LLaMA2-7B. We further train it on a large and diverse Arabic dataset, including all categories of Arabic, namely Classical Arabic (CA), Dialectal Arabic (DA), and MSA. This data is aggregated from various sources: AraNews_{v2} (Nagoudi et al., 2020), El-Khair (El-Khair, 2016), Gigaword,[3] OSCAR (Suárez et al., 2019), OSIAN (Zeroual et al., 2019), Wikipedia Arabic, and Hindawi Books.[4] We also derived ArabicWeb22 (A) and (B) from the open source Arabic text 2022.[5]

[3] LDC Catalog Link
[4] OpenITI corpus (v1.6) (Nigst et al., 2020).
[5] ArabicText-2022 data

Training Strategy. LLaMA-2's original vocabulary wasn't specifically optimized for Arabic and had a limited selection of Arabic words (it includes only 28 Arabic alphabet), leading to inadequate comprehension of Arabic language data. The initial step to address this involved expanding LLaMA-2's vocabulary. Expanding the vocabulary significantly enhanced the efficiency of encoding string sequences and enriched these sequences with more meaningful information, which was especially beneficial for document-level understanding and encoding (Li et al., 2023g). However, the limited amount of data available for continual pre-training meant that a substantial increase in vocabulary could introduce words or phrases that lack practical significance, posing a challenge in learning them effectively from the pre-training dataset and negatively affecting the model's performance. Moreover, a larger vocabulary size would increase the number of embedding-related parameters, which could impact training efficiency. Therefore, after conducting numerous experiments and considering the balance between training quality and efficiency, we increased the vocabulary size from the original 32,000 words in LLaMA-2 to 60,000 for the Arabic version. With the expanded vocabulary, the next step involved initializing the embeddings for the new vocabulary based on the original LLaMA-2 (7B) model. To ensure a smooth transition of capabilities from the original LLaMA-2 to the Arabic LLaMA-2 while keeping the English proficiency unaffected in the initial phase, we used a mean initialization method for the new embeddings, utilizing the weights from the original LLaMA-2. This approach not only preserved the English language capabilities but also facilitated the effective transfer of these capabilities to the Arabic language model, enabling LLaMA-2 to function efficiently in both English and Arabic. Moreover, 30 GB of English and Arabic data, was utilized during pre-training.

Instruction Finetuning. To enhance capabilities of our AraLLaMA we instruct-tuning it on three datasets: Alpaca-GPT4, Evol-instruct, ShareGPT extracted from MultilingualSIFT datasets (Chen et al.).

A.5 Models Parameters

In Table A.6, we provide details of the number of parameters for the main components of the Peacock models, including the distinction between trainable and non-trainable parameters for both the pertaining and instruction finetuning stages. These distinctions are vital for understanding the model's structure and optimization strategy.

Model	LLM	Vision Encoder	Q-Former	Projection Layer(s)	LLM	Total
InstructBlip	AraLLaMA	986M ❄	186M ❄	3M ✓✧	6.968B ✧	8.142B
InstructBlip	AceGPT	986M ❄	186M ❄	3M ✓✧	6.738B ✧	7.913B
LLaVA	AraLLaMA	304M ❄	—	21M ✓	6.967B ✧	7.292B
LLaVA	AceGPT	304M ❄	—	21M ✓	6.738B ✧	7.063B

Table A.6: Number of parameters for the main components of the models. ❄denotes frozen parameters during the pretraining and finetuning stages, ✓indicates trainable parameters trained from scratch in the pretraining stage, and ✧represents LoRA finetuning parameters in the instruction finetuning stage.

A.6 Henna Generation Pipeline and Examples

Henna was developed to establish a standard for evaluating VLMs in Arabic, aiming to incorporate elements related to Arabic culture. To achieve this, we selected images from Wikipedia and corresponding articles to create the context for GPT-4 during the generation process. The images were chosen to 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 the following categories: traditional food and cuisine, local customs, historical monuments and sites, common activities and lifestyles, and architectural styles and notable buildings. Figure A.9 demonstrates selected examples from the dataset's images.

We selected an image for each attraction and used GPT-4 to generate ten questions. Each image is accompanied by its Wikipedia article to enrich the context with comprehensive descriptions when prompting GPT-4 for generation. 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 A.10 and Figure A.11 demonstrates four randomly selected images with a generated pair of a question and their answers.

The Henna evaluation method employs a process where GPT-4 acts as the evaluator. In this approach, each response is assessed based on four criteria: Helpfulness, *Relevance*, *Accuracy*, and *Level of Details*. Each criterion is rated on a scale from 1 to 10, with higher scores indicating better responses. 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 A.15 illustrates three different examples where GPT-4's evaluations were high as Figure A.12 shown, medium as Figure A.13 shown, and low as Figure A.14 shown, for two different models' responses. An example of the prompt used in the evaluation is shown in Figure A.16.

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.

A.7 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 A.21 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.

A.8 Case Study with Egyptian Dialect Details and Examples

Figure A.22 displays three randomly selected examples of question-answer pairs translated from MSA into the Egyptian dialect. Figure A.23 presents four examples—three from Henna and one from LLaVA-Bench—along with the correct MSA answers and responses from two models: InstructBlip-AraLLaMA and InstructBlip-AceGPT, demonstrating the Peacock model's proficiency in responding in the Egyptian dialect. In Figure A.24, we highlight the superiority of Peacock over GPT-4 when posed with questions in the Egyptian dialect, where GPT-4 responds in MSA while Peacock provides answers in the Egyptian dialect. Conversely, when both models are queried in MSA, they reply in MSA.

Figure A.9: This collection of images showcases a curated subset selected from Henna dataset, representing 11 Arab countries, and capturing the essence of traditional food, local customs, historical monuments, everyday activities, and distinctive architecture that characterize the diverse and rich heritage of each region.

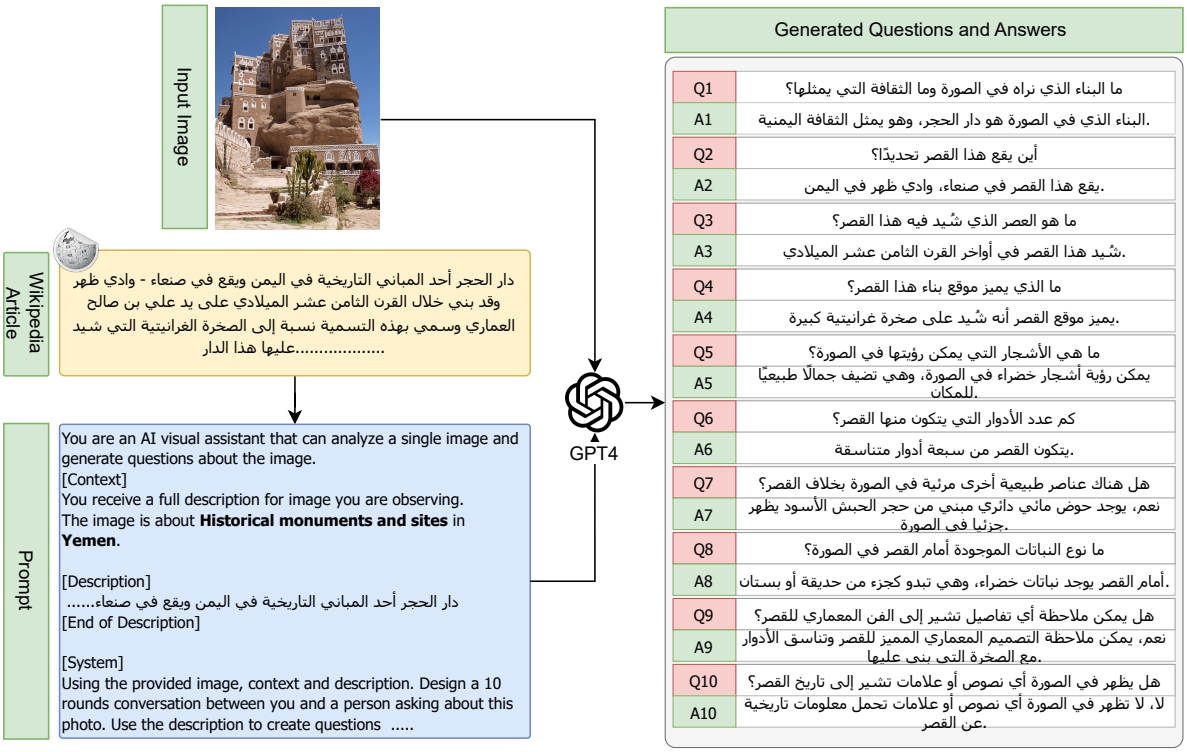

Figure A.10: Dataset Generation Example using GPT-4. This figure demonstrates the process of generating a question-answer dataset for an attraction in Yemen as an example. For each site, an image and its corresponding Wikipedia article were used to provide GPT-4 with rich contextual information. The model then generated ten contextually relevant questions and answers per image.

Q1	ما هو الاسم الذي يُعرف به هذا المكان وما الذي يميزه؟
A1	يُعرف هذا المكان بـشِبَام حَضْرَمَوْت وهو مشهور بأبراجه الطينية العالية، والتي يسميها يُلقب بـ'مانهاتن الصحراء'.
Q2	ما العناصر البارزة في هذه الصورة؟
A2	العناصر البارزة هي الأبنية الطينية الشاهقة التي تظهر بكثافة، وتلك هي الخصائص المعمارية الفريدة لشِبَام حَضْرَمَوْت.
Q3	أين يقع شِبَام حَضْرَمَوْت جغرافيًا؟
A3	يقع في المنطقة الوسطى الغربية من محافظة حضرموت، في صحراء رملة السبعتين.
Q4	ما هي الظروف المناخية التي يمكن استنتاجها من الصورة؟
A4	يمكن استنتاج أن المنطقة تتمتع بمناخ صحراوي حار نظرًا للرمال المحيطة والندرة الواضحة للغطاء النباتي.
Q5	ما هو التاريخ الذي يمكن ملاحظته في العمارة الموجودة بالصورة؟
A5	العمارة تعكس تاريخ طويل وعريق، حيث أن البناء الطيني المرتفع يدل على استخدام تقنيات بناء تقليدية قديمة.
Q6	هل يمكن رؤية آثار الحضارة الحديثة في هذه الصورة؟
A6	لا تظهر آثار الحضارة الحديثة بشكل واضح في الصورة، حيث أن الأبنية تحتفظ بطابعها التقليدي القديم.
Q7	هل يظهر تأثير الأحداث الحديثة، مثل الحروب، على الأبنية في الصورة؟
A7	لا يمكن تحديد آثار مباشرة للحروب على الأبنية من خلال الصورة.
Q8	هل تُظهر الصورة أي طرق أو مسالك مرورية؟
A8	نعم، يمكن ملاحظة وجود طرق ترابية تعبر بين الأبنية وحولها.
Q9	ما نوع الأشجار الموجودة في الصورة، إذا كانت موجودة؟
A9	يمكن رؤية بعض النخيل متفرقة، مما يشير إلى نوعية الغطاء النباتي المقتصر على أشجار قادرة على التكيف مع البيئة الصحراوية.
Q10	هل يمكن رؤية أي مظاهر للحياة اليومية في هذه الصورة؟
A10	الصورة لا تظهر مظاهر حيوية واضحة للحياة اليومية مثل الأشخاص أو النشاطات اليومية، فقط المباني والطرق.

Q1	ما نوع الهيكل الأساسي الذي يظهر في هذه الصورة؟
A1	الهيكل الأساسي الظاهر في الصورة هو تمثال لأبو الهول، وهو تمثال أسطوري يجسم أسد ورأس إنسان.
Q2	أين يقع هذا التمثال؟
A2	يقع هذا التمثال في هضبة الجيزة على الضفة الغربية من النيل في مدينة الجيزة، مصر.
Q3	ما هي العناصر المرئية الأخرى التي يمكن ملاحظتها في الخلفية؟
A3	في الخلفية يمكن ملاحظة أهرامات الجيزة، وهي من أشهر الآثار المصرية القديمة وتعتبر من عجائب الدنيا السبع.
Q4	ما الثقافة التي ينتمي إليها أبو الهول؟
A4	أبو الهول ينتمي إلى الثقافة المصرية القديمة، وهو يمثل الحضارة الفرعونية.
Q5	كم يبلغ طول تمثال أبو الهول وما هي أبعاده؟
A5	يبلغ طول تمثال أبو الهول حوالي 73.5 متر، وعرضه 19.3 متر، وارتفاعه حوالي 20 مترًا إلى قمة الرأس.
Q6	هل يوجد أي دليل على ألوان أصلية على التمثال في هذه الصورة؟
A6	لا يمكن ملاحظة آثار الألوان الأصلية على التمثال من خلال هذه الصورة.
Q7	ما الفترة التاريخية التي يعود إليها بناء أبو الهول؟
A7	يعود بناء أبو الهول إلى حوالي 2500 ق.م، ويُعتقد أنه تم بناؤه في عهد الملك خفرع.
Q8	ا هو الاتجاه الذي ينظر إليه تمثال أبو الهول؟
A8	تمثال أبو الهول ينظر نحو الشرق.
Q9	هل هناك أي تشوهات مرئية أو تلف في جسم التمثال؟
A9	نعم، يمكن ملاحظة التآكل والتشوهات في جسم التمثال، خاصة حول الوجه والجسم نتيجة لعوامل التعرية على مر الزمن.
Q10	ما هو الدور الذي كان يقوم به أبو الهول في الحضارة المصرية القديمة؟
A10	كان أبو الهول يقوم بدور الحارس لهضبة الجيزة، حيث يقع بالقرب من الأهرامات ويعتبر رمزًا للقوة والحماية.

Q1	ما نوع الطبق المعروض في الصورة؟
A1	الطبق المعروض في الصورة يبدو أنه من الحلويات الشرقية، وهو يحتوي على قطع مربعة الشكل تشبه حلوى قلب اللوز.
Q2	أين يتم وضع هذا الطبق عادةً؟
A2	يبدو أن هذا الطبق يتم تقديمه في المناسبات الخاصة والأعياد، كما يظهر أنه مقدم على صينية مزخرفة تشير إلى أهميته.
Q3	ما المكونات الظاهرة على سطح الحلوى؟
A3	على سطح الحلوى يمكن ملاحظة وجود حبات من اللوز أو المكسرات المشابهة.
Q4	ما الذي يحيط بقطع الحلوى في الصورة؟
A4	يحيط بقطع الحلوى مسحوق أخضر اللون، والذي يمكن أن يكون مسحوق الفستق الحلبي.
	.....
	.....

Q1	ما الفعالية التي يُشارك بها هؤلاء الأشخاص في الصورة؟
A1	يبدو أن الأشخاص في الصورة يُشاركون في فعالية موسيقية تقليدية،
Q2	ما نوع الآلات الموسيقية المستخدمة في الصورة؟
A2	في الصورة تظهر آلات موسيقية مختلفة منها العود والدربوكة والكمان وآلة تشبه الأكورديون المغربية.
Q3	أين يجلس العازفون وما الذي يحيط بهم؟
A3	يجلس العازفون على كراسي وعلى الأرض، ويحيط بهم جدران مزينة بالفسيفساء والزليج المغربي الذي يعكس غنى الثقافة المغربية.
Q4	ما الزي الذي يرتديه الأشخاص في الصورة؟
A4	الأشخاص في الصورة يرتدون الجلابة المغربية التقليدية، ويضعون على رؤوسهم الطربوش الأحمر، وهو زي تقليدي مغربي.
....	.....
.....	.....

Figure A.11: Demonstration of four randomly selected images, each accompanied by question and answer pairs generated by GPT-4.

Figure A.12: High-scoring evaluations: The answer from InstructBlip-AraLLaMA is accurate, earning a perfect score from GPT-4. mBlip's response is also rated as reasonable.

Figure A.13: Moderate-scoring evaluations: Here, InstructBlip-AraLLaMA's response, while accurate, doesn't match the gold standard, resulting in a moderate GPT-4 score. mBlip's scoring reflects a proper evaluation.

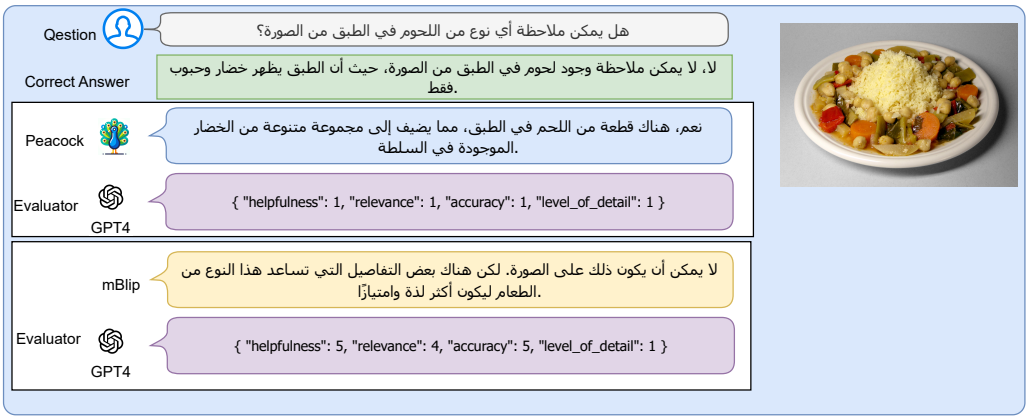

Figure A.14: Low-scoring evaluations: InstructBlip-AraLLaMA's response is incorrect, leading to a low GPT-4 score. Conversely, mBlip is awarded a moderate score, indicating a more accurate response.

Figure A.15: Henna evaluation samples showing GPT-4's scoring range for responses from different models. Subfigures illustrate the spectrum of scoring from high to low, based on the accuracy and relevance of the model-generated answers to the benchmark questions.

Given the following question in Arabic: {question}, and the correct answer in Arabic: {correct_answer}, please rate the following aspects of the model's response: helpfulness, relevance, accuracy, and level of detail.
Rate and score each aspect on a scale of 1 to 10, where a higher score indicates better overall quality of the model's response.
The model's response is enclosed within triple backticks.
Format your feedback as a JSON object, using 'helpfulness', 'relevance', 'accuracy', and 'level_of_details' as keys.
If the information is not available, use "unknown" as the value.

Model response: '''{model_response}'''

Figure A.16: The prompt used to guide ChatGPT to evaluate Henna.

Figure A.17: InstructBlip-AraLLaMA Examples.

Figure A.18: InstructBlip-AceGPT Examples.

Figure A.19: LLaVA-AraLLaMA Examples.

Figure A.20: LLaVA-AceGPT Examples.

Figure A.21: Selected examples from the LLaVA benchmark include two from the conversational type, two from the detail type, and two from the complex question type. The top-left responses are from InstructBlip-AraLLaMA, the top-right responses from InstructBlip-AceGPT, the bottom-left responses from LLaVA-AraLLaMA, and the bottom-right responses from LLaVA-AceGPT.

Question in MSA	Question in Egyptian Dialect	Answer in MSA	Answer in Egyptian Dialect
ما الذي يخلق مظهر الأضواء المتقطعة في الصورة؟	ايه اللي بيخلق شكل الاضواء المتقطعة في الصورة؟	يتم إنشاء مظهر الأضواء المتقطعة في الصورة من خلال تقنية التصوير الفوتوغرافي بفاصل زمني، والتي تلتقط حركة السيارات على أنها ضبابية أثناء سفرها عبر شارع المدينة.	بيتم عمل شكل للانوار المتقطعة من خلال تقنية التصوير الفتوغرافي بفاصل زمني واللي بتلقط حركة العربيات على إنها ضبابية في وقت سفرها جوا المدينة
ماذا يرتدي الرجل؟	الراجل ده لابس ايه؟	يرتدي الرجل قميصًا أبيض وسروالًا قصيرًا أثناء لعب التنس.	الراجل لابس قميص أبيض وشورت قصير وهو بيلعب التنس.
هل هذه الصورة قديمة؟	الصورة دي قديمة؟	نعم	أيوه

Figure A.22: Three examples of question-answer pairs translated from MSA to the Egyptian dialect.

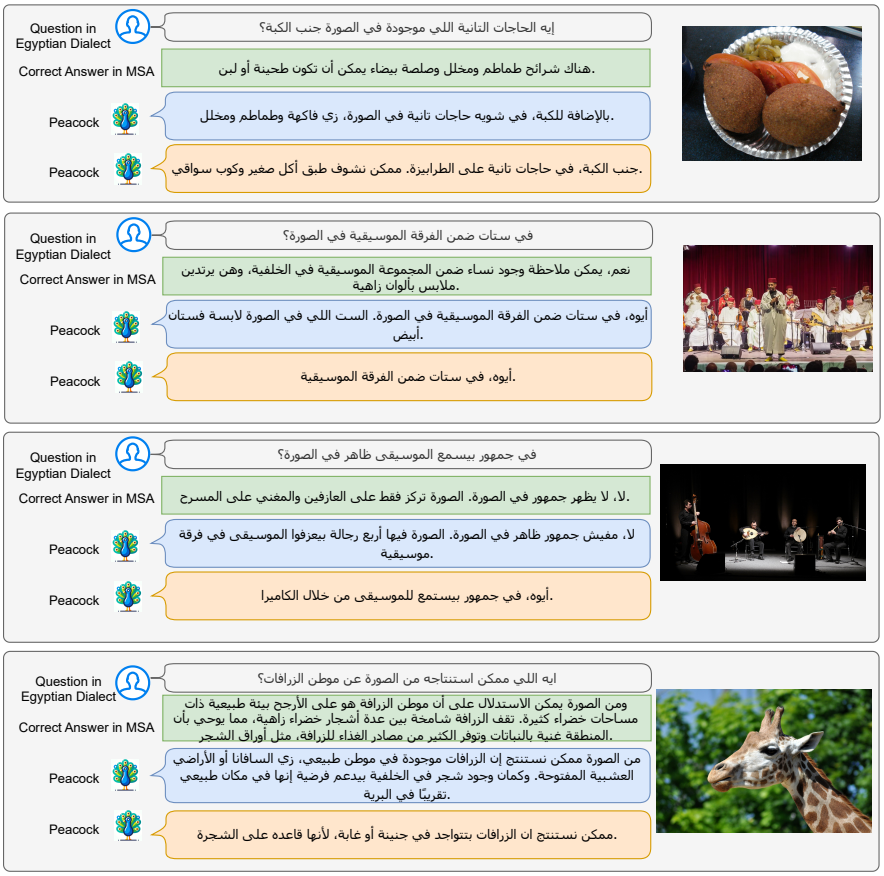

Figure A.23: Comparative responses from InstructBlip-AraLLaMA and InstructBlip-AceGPT models alongside the correct MSA answers for four selected examples, three from Henna and the last one from LLaVA-Bench, demonstrating the models' proficiency in Egyptian dialect.

Figure A.24: A demonstration of Peacock model's advantage over GPT-4 in responding to queries in the Egyptian dialect, with the former responding in the dialect and the latter in MSA. A secondary example shows both models replying in MSA when queried in MSA.