A Broader Impact

InstructBLIP uses off-the-shelf frozen LLMs. Therefore it inherits some of the shortcomings from the original LLMs, such as hallucinating ungrounded text or generating outputs with bias. We mitigate such shortcomings by improving the model’s grounding on the vision and instruction input, and performing vision-language instruction tuning on a diverse set of high-quality datasets. Nevertheless, we do not recommend applying InstructBLIP models to any downstream applications without a prior assessment on safety and fairness specific to that application.

B More Case Studies

Figure 1: The response from InstructBLIP is more comprehensive than GPT-4, more visually-grounded than LLaVA, and more logical than MiniGPT-4. The responses of GPT-4 and LLaVA are obtained from their respective papers, while the official demo is used for MiniGPT-4. Source: https://www.barnorama.com/wp-content/uploads/2016/12/03-Confusing-Pictures.jpg
The painting depicts a woman, commonly believed to be Mona Lisa, the famous artwork by Leonardo da Vinci. It is a portrait painting that showcases the woman's enigmatic smile and has become one of the most famous and iconic art pieces in the world. The original work is displayed in the Louvre Museum in Paris, and it is known for its intricate details, use of oil paint, and the artist's innovative techniques that contributed to its enduring appeal and mystery.

The painting is the Mona Lisa, also known as the Mona Lisa, by Leonardo da Vinci. It is one of the most famous paintings in the world and is known for its enigmatic smile and the mysterious identity of the sitter. The painting is housed in the Louvre Museum in Paris, France.

Figure 2: In this example, we illustrate that long-form responses are not always preferable. InstructBLIP can directly address the user’s intent by adaptively adjusting the response length, while other models tend to generate lengthy paragraphs with less-relevant sentences. The response from LLaVA is taken from the paper, and for MiniGPT-4, we utilize its official demo.
The painting depicts a young girl with a pearl earring, also known as "The Girl with the Pearl Earring" or "Girl with a Pearl Earring". She is wearing a blue dress and has a pearl earring on her left ear. Her hair is pulled back into a bun, and she is looking directly at the viewer. This painting is a famous work of art by Dutch painter Johannes Vermeer, created in 1665. It is considered one of his masterpieces and is widely recognized for its intricate details and realistic portrayal of the young girl’s facial expression.

The painting features a woman with a headscarf, possibly of Dutch origin, dressed in blue and white clothing. She is wearing a large blue hat and has a red mouth, which adds a vibrant touch to the portrait. The woman is looking directly at the viewer, creating a sense of connection. The background of the painting is predominantly blue, further emphasizing the color theme of the piece. The overall composition and attire of the woman suggest that the painting might be inspired by the style of Rembrandt.

The painting is well-executed, with a high level of detail and realism in the depiction of the woman’s features and clothing. The use of shading and lighting creates a sense of depth and dimension, making the woman appear three-dimensional. The artist has used a limited color palette, with the woman’s dress and apron providing the only sources of color in the painting.

The painting is well-composed, with the woman’s head and shoulders positioned in the center of the canvas, creating a sense of balance and symmetry. The use of negative space around the woman’s figure adds to the overall somber mood of the painting.

Overall, this is a well-executed and thought-provoking painting that effectively conveys a sense of contemplation and introspection.

Figure 3: InstructBLIP shows the ability to connect textual knowledge with visual input, while the other models depict the picture plainly. MiniGPT-4 exhibits poorer results, which may be due to its training with only long captions. Responses of LLaVA and MiniGPT-4 are generated by their official demos.
### Instruction Tuning Datasets

<table>
<thead>
<tr>
<th>Dataset Name</th>
<th>Held-out</th>
<th>Dataset Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COCO Caption [1]</td>
<td>✗</td>
<td>We use the large-scale COCO dataset for the image captioning task. Specifically, Karpathy split [2] is used, which divides the data into 82K/5K/5K images for the train/val/test sets.</td>
</tr>
<tr>
<td>Web CapFilt</td>
<td>✗</td>
<td>14M image-text pairs collected from the web with additional BLIP-generated synthetic captions, used in BLIP [3] and BLIP-2 [4].</td>
</tr>
<tr>
<td>Flickr30K [6]</td>
<td>✗</td>
<td>The Flickr30k dataset consists of 31K images collected from Flickr, each image has five ground truth captions. We use the test split as the held-out which contains 1K images.</td>
</tr>
<tr>
<td>TextCaps [7]</td>
<td>✓ (test)</td>
<td>TextCaps is an image captioning dataset that requires the model to comprehend and reason the text in images. Its train/val/test sets contain 21K/3K/3K images, respectively.</td>
</tr>
<tr>
<td>VQAv2 [8]</td>
<td>✗</td>
<td>VQAv2 is dataset for open-ended image question answering. It is split into 82K/40K/81K for train/val/test.</td>
</tr>
<tr>
<td>VizWiz [9]</td>
<td>✓ (test-dev)</td>
<td>A dataset contains visual questions asked by people who are blind. 8K images are used for the held-out evaluation.</td>
</tr>
<tr>
<td>GQA [10]</td>
<td>✓ (test-dev)</td>
<td>GQA contains image questions for scene understanding and reasoning. We use the balanced test-dev set as held-out evaluation.</td>
</tr>
<tr>
<td>Visual Spatial Reasoning</td>
<td>✓ (test)</td>
<td>VSR is a collection of image-text pairs, in which the text describes the spatial relation of two objects in the image. Models are required to classify true/false for the description. We use the zero-shot data split given in its official github repository.</td>
</tr>
<tr>
<td>IconQA [11]</td>
<td>✓ (test)</td>
<td>IconQA measures the abstract diagram understanding and comprehensive cognitive reasoning abilities of models. We use the test set of its multi-text-choice task for held-out evaluation.</td>
</tr>
<tr>
<td>OKVQA [12]</td>
<td>✗</td>
<td>OKVQA contains visual questions that require outside knowledge to answer. It has been split into 9K/5K for train and test.</td>
</tr>
<tr>
<td>A-OKVQA [13]</td>
<td>✗</td>
<td>A-OKVQA is a successor of OKVQA with more challenging and diverse questions. It has 17K/1K/6K questions for train/val/test.</td>
</tr>
<tr>
<td>ScienceQA [14]</td>
<td>✓ (test)</td>
<td>ScienceQA covers diverse science topics with corresponding lectures and explanations. In out settings, we only use the part with image context (IMG).</td>
</tr>
<tr>
<td>Visual Dialog [15]</td>
<td>✓ (val)</td>
<td>Visual dialog is a conversational question answering dataset. We use the val split as the held-out, which contains 2,064 images and each has 10 rounds.</td>
</tr>
<tr>
<td>OCR-VQA [16]</td>
<td>✗</td>
<td>OCR-VQA contains visual questions that require models to read text in the image. It has 800K/100K/100K for train/val/test, respectively.</td>
</tr>
<tr>
<td>TextVQA [17]</td>
<td>✓ (val)</td>
<td>TextVQA requires models to comprehend visual text to answer questions.</td>
</tr>
<tr>
<td>HatefulMemes [18]</td>
<td>✓ (val)</td>
<td>A binary classification dataset to justify whether a meme contains hateful content.</td>
</tr>
<tr>
<td>LLaVA-Instruct-150K [19]</td>
<td>✗</td>
<td>An instruction tuning dataset which has three parts: detailed caption (23K), reasoning (77K), conversation (50K).</td>
</tr>
<tr>
<td>MSVD-QA [20]</td>
<td>✓ (test)</td>
<td>We use the test set (13K video QA pairs) of MSVD-QA for held-out testing.</td>
</tr>
<tr>
<td>MSRVTT-QA [20]</td>
<td>✓ (test)</td>
<td>MSRVTT-QA has more complex scenes than MSVD, with 72K video QA pairs as the test set.</td>
</tr>
<tr>
<td>iVQA [21]</td>
<td>✓ (test)</td>
<td>iVQA is a video QA dataset with mitigated language biases. It has 6K/2K/2K samples for train/val/test.</td>
</tr>
</tbody>
</table>

Table 1: Description of datasets in our held-in instruction tuning and held-out zero-shot evaluations.
D Instruction Templates

<table>
<thead>
<tr>
<th>Task</th>
<th>Instruction Template</th>
</tr>
</thead>
</table>
| Image Captioning | <Image>A short image caption:  
<Image>A short image description:  
<Image>A photo of  
<Image>An image that shows  
<Image>Write a short description for the image.  
<Image>Write a description for the photo.  
<Image>Provide a description of what is presented in the photo.  
<Image>Briefly describe the content of the image.  
<Image>Can you briefly explain what you see in the image?  
<Image>Could you use a few words to describe what you perceive in the photo?  
<Image>Please provide a short depiction of the picture.  
<Image>Using language, provide a short account of the image.  
<Image>Use a few words to illustrate what is happening in the picture. |
| VQA | <Image>{Question}  
<Image>Question: {Question}  
<Image>{Question} A short answer to the question is  
<Image>Q: {Question} A:  
<Image>Question: {Question} Short answer:  
<Image>Given the image, answer the following question with no more than three words. {Question}  
<Image>Answer: {Question}.  
<Image>Use the provided image to answer the question: {Question} Provide your answer as short as possible:  
<Image>What is the answer to the following question? "{Question}"  
<Image>The question \"{Question}\" can be answered using the image. A short answer is |
| VQG | <Image>Given the image, generate a question whose answer is: \{Answer\}. Question:  
<Image>Based on the image, provide a question with the answer: \{Answer\}. Question:  
<Image>Given the visual representation, create a question for which the answer is \"\{Answer\}\".  
<Image>From the image provided, craft a question that leads to the reply: \{Answer\}. Question:  
<Image>Considering the picture, come up with a question where the answer is: \{Answer\}.  
<Image>Taking the image into account, generate an question that has the answer: \{Answer\}. Question: |

Table 2: Instruction templates used for transforming held-in datasets into instruction tuning data. For datasets with OCR tokens, we simply add “OCR tokens:” after the image query embeddings.

References


