MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models

Deyao Zhu* **Jun Chen*** **Xiaoqian Shen** **Xiang Li** **Mohamed Elhoseiny**
King Abdullah University of Science and Technology
{deyao.zhu, jun.chen, xiaoqian.shen, xiang.li.1, mohamed.elhoseiny}@kaust.edu.sa

Abstract

The recent GPT-4 has demonstrated extraordinary multi-modal abilities, such as directly generating websites from handwritten text and identifying humorous elements within images. These features are rarely observed in previous vision-language models. We believe the primary reason for GPT-4's advanced multi-modal generation capabilities lies in the utilization of a more advanced large language model (LLM). To examine this phenomenon, we present MiniGPT-4, which aligns a frozen visual encoder with a frozen LLM, Vicuna, using just one projection layer. Our findings reveal that MiniGPT-4 possesses many capabilities similar to those exhibited by GPT-4 like detailed image description generation and website creation from hand-written drafts. Furthermore, we also observe other emerging capabilities in MiniGPT-4, including writing stories and poems inspired by given images, providing solutions to problems shown in images, teaching users how to cook based on food photos, etc. In our experiment, we found that only performing the pretraining on raw image-text pairs could produce unnatural language outputs that lack coherency including repetition and fragmented sentences. To address this problem, we curate a high-quality, well-aligned dataset in the second stage to finetune our model using a conversational template. This step proved crucial for augmenting the model's generation reliability and overall usability. Notably, our model is highly computationally efficient, as we only train a projection layer utilizing approximately 5 million aligned image-text pairs. Our code, pre-trained model, and collected dataset are available at `https://minigpt-4.github.io/`.

1 Introduction

In recent years, large language models (LLMs) have experienced rapid advancements [21, 18, 4, 24, 32, 9, 14]. With exceptional language understanding capabilities, these models can perform a variety of intricate linguistic tasks in a zero-shot manner. Notably, GPT-4 [19], a large-scale multimodal model, has been recently introduced with demonstrating many impressive capabilities. For example, GPT-4 can produce very detailed and accurate image descriptions, explain unusual visual phenomena, and even construct websites based on handwritten text instructions.

Although GPT-4 has exhibited remarkable capabilities, the methods behind its exceptional abilities are still a mystery [19]. We believe that these superior skills may stem from the utilization of a more advanced large language model (LLM). LLMs have demonstrated various emergent abilities, as evidenced in GPT-3's few-shot prompting setup [4] and the findings of Wei *et al.* (2022) [34]. Such emergent properties are hard to find in smaller-scale models. It is conjectured that these emergent

*Equal contribution

Preprint. Work in progress.

abilities are also applicable to multi-modal models, which could be the foundation of GPT-4's impressive visual description capabilities.

To substantiate our hypothesis, we present a novel model named MiniGPT-4. It utilizes an advanced large language model (LLM), Vicuna [8], which is built upon LLaMA [32] and reported to achieve 90% of ChatGPT's quality as per GPT-4's evaluation, as the language decoder. In terms of visual perception, we employ the same pretrained vision component of BLIP-2 [16] that consists of a ViT-G/14 from EVA-CLIP [13] and a Q-Former. MiniGPT-4 adds a single projection layer to align the encoded visual features with the Vicuna language model and freezes all the other vision and language components. MiniGPT-4 is initially trained for 20k steps using a batch size of 256 on 4 A100 GPUs, leveraging a combined dataset that includes images from LAION [26], Conceptual Captions [5, 27], and SBU [20] to align visual features with the Vicuna language model. However, simply aligning the visual features with the LLM is insufficient to train high-performing model with visual conversation abilities like a chatbot, and the noises underlying the raw image-text pairs may result in incoherent language output. Therefore, we collect another 3,500 high-quality aligned image-text pairs to further fine-tune the model with a designed conversational template in order to improve the naturalness of the generated language and its usability.

In our experiments, we discovered that MiniGPT-4 possesses numerous capabilities similar to those demonstrated by GPT-4. For instance, MiniGPT-4 can generate intricate image descriptions, create websites based on handwritten text instructions, and explain unusual visual phenomena. Furthermore, our findings revealed that MiniGPT-4 also has a variety of other intriguing abilities not showcased in the GPT-4 demonstrations. For example, MiniGPT-4 can directly generate detailed recipes by observing appetizing food photos, craft stories or rap songs inspired by images, write advertisements for products in images, distinguish problems shown in photos and provide corresponding solutions, and retrieve rich facts about people, movies, or art directly from images, among other capabilities. These abilities are absent in previous vision-language models like Kosmos-1 [15] and BLIP-2 [16], which do not apply a stronger language model such as Vicuna. This contrast validates that integrating visual features with an advanced language model can yield emergent vision-language abilities.

We present a summary of our key findings:

- Our research reveals that by aligning visual features with the advanced large language model, Vicuna, we can achieve emergent vision-language capabilities. We demonstrate that our MiniGPT-4 can process abilities similar to those showcased in the GPT-4 demonstrations.
- By utilizing a pretrained vision encoder and a large language model, MiniGPT-4 achieves greater computational efficiency. Our findings suggest that training merely one projection layer can effectively align the visual features with the large language model. Our MiniGPT-4 only requires training approximately 10 hours on 4 A100 GPUs.
- We discovered that simply aligning visual features with large language models using raw image-text pairs from public datasets is not sufficient for developing a well-performing MiniGPT-4 model. It may produce unnatural language outputs that lack coherency including repetition and fragmented sentences. Addressing this limitation requires training with a high-quality, well-aligned dataset, which significantly improves its usability.

2 Related Works

Large language models have experienced tremendous success in recent years due to the scaling up of training data and an increase in the number of parameters. Early models, such as BERT [11], GPT-2 [22], and T5 [23], laid the foundation for this progress. Subsequently, GPT-3 [4], with a massive scale of 175 billion parameters, was introduced, demonstrating significant breakthroughs across numerous language benchmarks. This development inspired the creation of various other large language models, including Megatron-Turing NLG [28], Chinchilla [14], PaLM [9], OPT [38], BLOOM [25], and LLaMA [32], among others. Wei *et al.* [34] further discovered several *emergent abilities*, which appear exclusively in large models. The emergence of these abilities underscores the importance of scaling up in the development of large language models. Moreover, by aligning the pre-trained large language model GPT-3 with human intent, instructions and human feedback, InstructGPT [21] and ChatGPT [18] enable conversational interactions with humans and can answer a wide range of diverse and complex questions. More recently, several open-sourced models, such

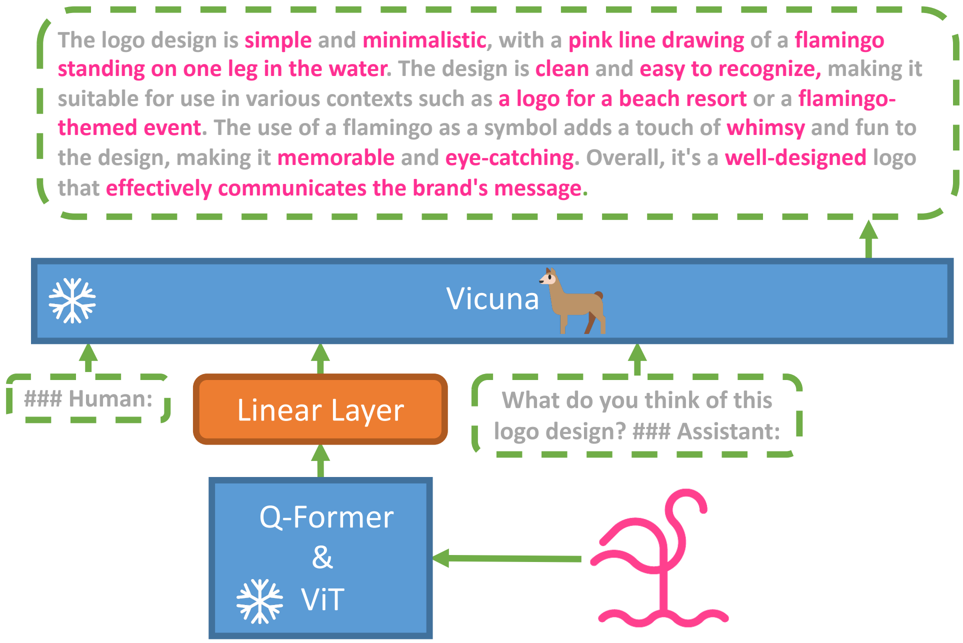


Figure 1: **The architecture of MiniGPT-4.** It consists of a vision encoder with a pretrained ViT and Q-Former, a single linear projection layer, and an advanced Vicuna large language model. MiniGPT-4 only requires training the linear projection layer to align the visual features with the Vicuna.

as Alpaca [30] and Vicuna [8], have been developed based on LLaMA [32] and also exhibit similar performance.

Leveraging Pre-trained LLMs in Vision-Language Tasks. In recent years, the trend of using autoregressive language models as decoders in vision-language tasks has gained significant traction [6, 15, 36, 31, 2, 16, 17, 12]. This approach takes advantage of cross-modal transfer, allowing knowledge to be shared between language and multimodal domains. Pioneering studies like VisualGPT [6] and Frozen [33] have demonstrated the benefits of employing a pre-trained language model as a vision-language model decoder. Flamingo [2] was then developed to align a pre-trained vision encoder and language model using gated cross-attention, and was trained on billions of image-text pairs, showcasing impressive in-context few-shot learning capabilities. Following that, BLIP-2 [16] was introduced, employing a Flan-T5 [10] with a Q-Former to efficiently align visual features with the language model. Most recently, PaLM-E [12], featuring 562 billion parameters, has been developed to integrate real-world continuous sensor modalities into an LLM, thereby establishing a connection between real-world perceptions and human languages. GPT-4 [19] has also been recently released, showcasing more powerful visual understanding and reasoning abilities after pre-training on a vast collection of aligned image-text data.

LLMs, such as ChatGPT, have proven to be powerful tools in enhancing the performance of vision-language tasks by collaborating with other specialized models. For instance, Visual ChatGPT [35] and MM-REACT [37] showcase how ChatGPT can act as a coordinator, integrating with diverse visual foundation models and facilitating their collaboration to tackle more complex challenges. ChatCaptioner [39] treats ChatGPT as a questioner, prompting diverse questions for BLIP-2 to answer. Through multi-round conversations, ChatGPT extracts visual information from BLIP-2 and effectively summarizes the image content. Video ChatCaptioner [7] extends this approach, applying it to video spatiotemporal understanding. ViperGPT [29] demonstrates the potential of combining an LLM with different vision models to address complex visual queries programmatically. In contrast, MiniGPT4 directly aligns visual information with the language model to accomplish diverse vision-language tasks without the usage of external vision models.

3 Method

MiniGPT-4 aims to align visual information from a pretrained vision encoder with an advanced large language model (LLM). Specifically, we utilize the Vicuna [8] as our language decoder, which is constructed upon LLaMA [32] and can perform a wide range of complex linguistic tasks. For visual perception, we employ the same visual encoder as used in BLIP-2 [16], a ViT backbone [13] coupled with their pre-trained Q-Former. Both language and vision models are open-sourced. We target to bridge the gap between the visual encoder and LLM using a linear projection layer, with an overview of our model displayed in Fig.1.

To achieve an effective MiniGPT-4, we propose a two-stage training approach. The initial stage involves pretraining the model on a large collection of aligned image-text pairs to acquire vision-language knowledge. In the second stage, we fine-tune the pretrained model with a smaller but high-quality image-text dataset with a designed conversational template to enhance the model's generation reliability and usability.

3.1 First pretraining stage

During the initial pretraining stage, the model is designed to acquire vision-language knowledge from a large collection of aligned image-text pairs. We regard the output from the injected projection layer as a soft prompt for the LLM, prompting it to generate the corresponding ground-truth texts.

Throughout the entire pretraining process, both the pretrained vision encoder and the LLM remain frozen, with only the linear projection layer being pretrained. We use a combined dataset of Conceptual Caption [5, 27], SBU [20] and LAION [26] to train our model. Our model undergoes 20,000 training steps with a batch size of 256, covering approximately 5 million image-text pairs. The entire process takes about 10 hours to complete, utilizing 4 A100 (80GB) GPUs.

Issues of the first pretraining stage. Following the first pretraining stage, our MiniGPT-4 demonstrates the capacity to possess a wealth of knowledge and offer reasonable responses to human inquiries. However, we have observed instances where it struggles to produce coherent linguistic output, such as generating repetitive words or sentences, fragmented sentences, or irrelevant content. These issues hinder MiniGPT-4's ability to engage in a fluent visual conversation with humans.

We have also noticed that similar issues were also faced in GPT-3. Despite being pretrained on an extensive language dataset, GPT-3 could not directly generate language outputs that are in accordance with the users' intentions. Through a process of instruction fine-tuning and reinforcement learning from human feedback, GPT-3 evolves into GPT-3.5 [21, 18] and becomes capable of producing more human-friendly outputs. This phenomenon bears a resemblance to the current state of MiniGPT-4 following its initial pretraining stage. As such, it is not surprising that our model may struggle to generate fluent and natural human language outputs at this stage.

3.2 Curating a high-quality alignment dataset for vision-language domain.

To achieve greater naturalness in the generated language and enhance the model's usability, a second-stage alignment process is essential. While in the realm of NLP, instruction fine-tuning datasets [30] and conversations [1] are easily accessible, no equivalent datasets exist for the vision-language domain. To address this deficiency, we carefully curated a high-quality image-text dataset, specifically tailored for alignment purposes. This dataset is subsequently utilized to fine-tune our MiniGPT-4 during the second-stage alignment process.

Initial aligned image-text generation In the initial phase, we employ the model derived from the first pretraining stage to generate a comprehensive description of a given image. To enable our model to produce the more detailed image descriptions, we have designed a prompt that adheres to the conversational format of the Vicuna [8] language model, as shown below:

###Human: <Img><ImageFeature></Img> Describe this image in detail. Give as many details as possible. Say everything you see. ###Assistant:

In this prompt, *<ImageFeature>* represents the visual features produced by the linear projection layer.

To identify incomplete sentences, we examine whether the generated sentence exceeds 80 tokens. If it does not, we incorporate an additional prompt, *###Human: Continue ###Assistant:* , prompting our MiniGPT-4 to extend the generation. By concatenating the outputs from both steps, we can create a more comprehensive image description. This approach enables us to generate more image-text pairs with detailed and informative image descriptions. We randomly select 5,000 images from the Conceptual Caption dataset [5, 27] and employ this approach to generate corresponding language descriptions for each image.

Data post-processing The generated image descriptions still have much noises and contain the errors, such as repetition of words or sentences, and the presence of incoherent statements. In order to mitigate these issues, we employ ChatGPT to refine the descriptions by utilizing the subsequent prompt:

Fix the error in the given paragraph. Remove any repeating sentences, meaningless characters, not English sentences, and so on. Remove unnecessary repetition. Rewrite any incomplete sentences. Return directly the results without explanation. Return directly the input paragraph if it is already correct without explanation.

Upon completing the post-processing stage, we manually verify the correctness of each image description to guarantee its high quality. Specifically, we check if each generated image description follows our desired format, and also manually refine the generated captions by eliminating redundant words or sentences that ChatGPT fails to detect. Finally, only approximately 3,500 out of 5,000 image-text pairs satisfy our requirement, and these pairs are subsequently utilized for the second-stage alignment process.

3.3 Second-stage finetuning

During the second stage, we finetune our pretrained model with the curated high-quality image-text pairs. During the finetuning, we use the predefined prompts in the following template:

###Human: <Img><ImageFeature></Img> <Instruction> ###Assistant:

In this prompt, *<Instruction>* represents a randomly sampled instruction from our predefined instruction set containing variant forms of instructions such as *"Describe this image in detail"* or *"Could you describe the contents of this image for me"*. It is important to note that we do not calculate the regression loss for this specific text-image prompt.

As a result, MiniGPT-4 is now capable of producing more natural and reliable responses. Furthermore, we have observed that the model's fine-tuning process is remarkably efficient, only requiring a mere 400 training steps with batch size of 12, which takes a brief 7 minutes to complete with a single A100 GPU.

4 Demonstrations:

Our MiniGPT-4 exhibits a multitude of capabilities similar to those demonstrated by GPT-4. These include generating detailed image descriptions (Fig. 2), identifying amusing aspects within images (Fig. 3), and uncovering unusual content (Fig. 4). Additionally, the model can generate websites from handwritten text (Fig. 5). We have also discovered that our MiniGPT-4 possesses other abilities such as identifying problems in images and providing solutions (Fig. 6), creating poems or rap songs inspired by images (Fig. 7), writing stories for images (Fig. 8), making advertisements for products in images (Fig. 9), identifying individuals (Fig. 10), providing insightful image comments (Fig. 11), retrieving facts related to images (Fig. 12), and teaching users to cook foods with given photos (Fig. 13). These diverse examples showcase the strong capabilities of our MiniGPT-4.

5 Limitations

Although MiniGPT-4 processes numerous advanced vision-language capabilities, as displayed in our demonstrations, it currently still faces several limitations.

Language hallucination. As MiniGPT-4 is built upon LLMs, it inherits LLM's limitations like unreliable reasoning ability and hallucinating nonexistent knowledge. This issue might be alleviated

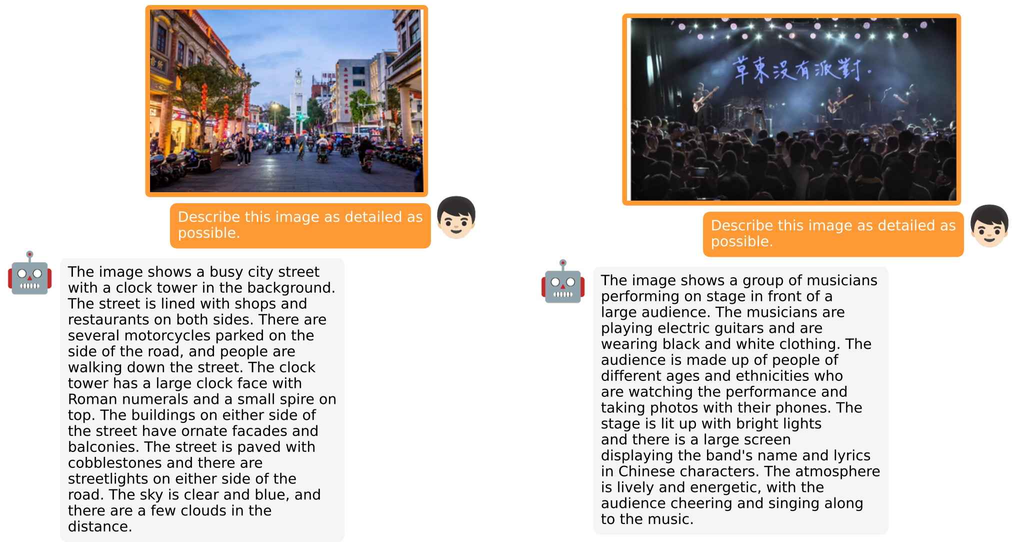


Figure 2: Detailed image descriptions

by training the model with more high-quality, aligned image-text pairs, or aligning with more advanced LLMs in the future.

Inadequate perception capacities. MiniGPT-4's visual perception remains limited. It may struggle to recognize detailed textual information from images, and differentiate spatial localization. This limitation may stem from several factors: 1) A lack of sufficient aligned image-text data containing adequate information such as spatial localization and optical character annotations. This issue could be alleviated by training on more well-aligned and rich data. 2) The frozen Q-former used in the visual encoder may lose some essential features, such as visual-spatial grounding. This could potentially be improved by replacing it with a stronger visual perception model. 3) Training only one projection layer might not provide enough capacity to learn extensive visual-text alignment.

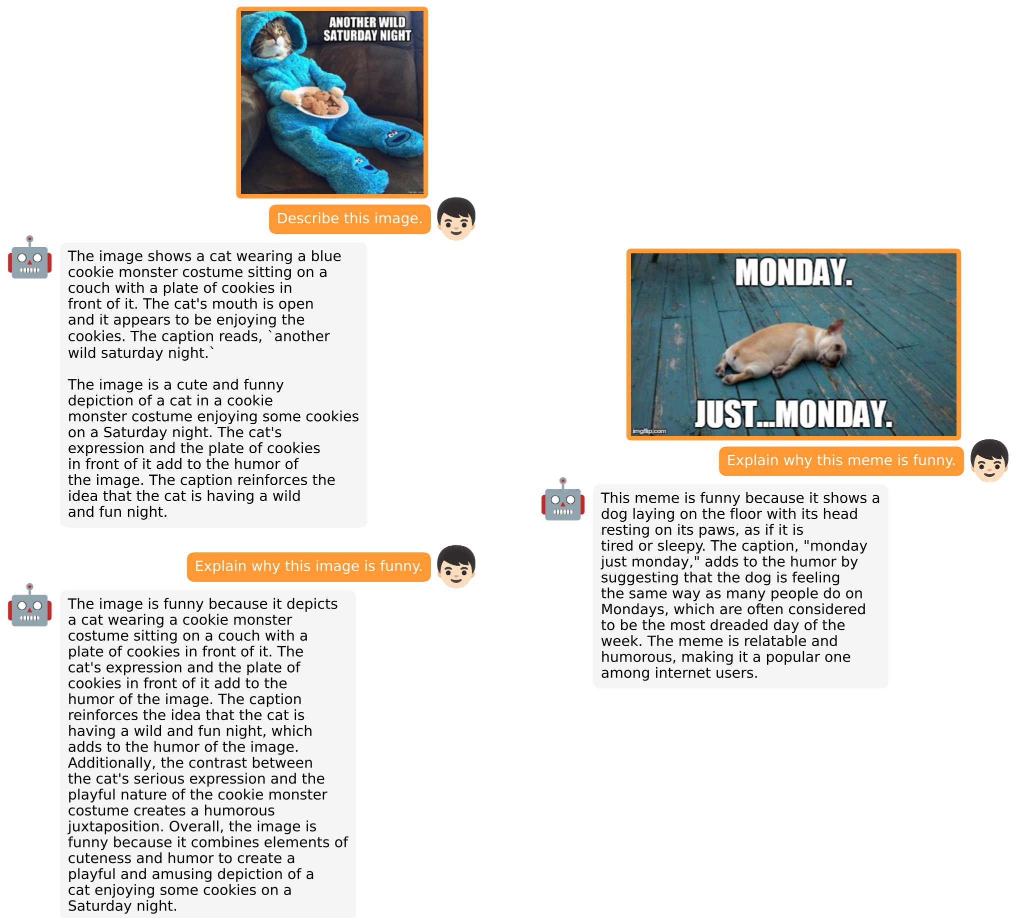


Figure 3: Identifying amusing aspects within images

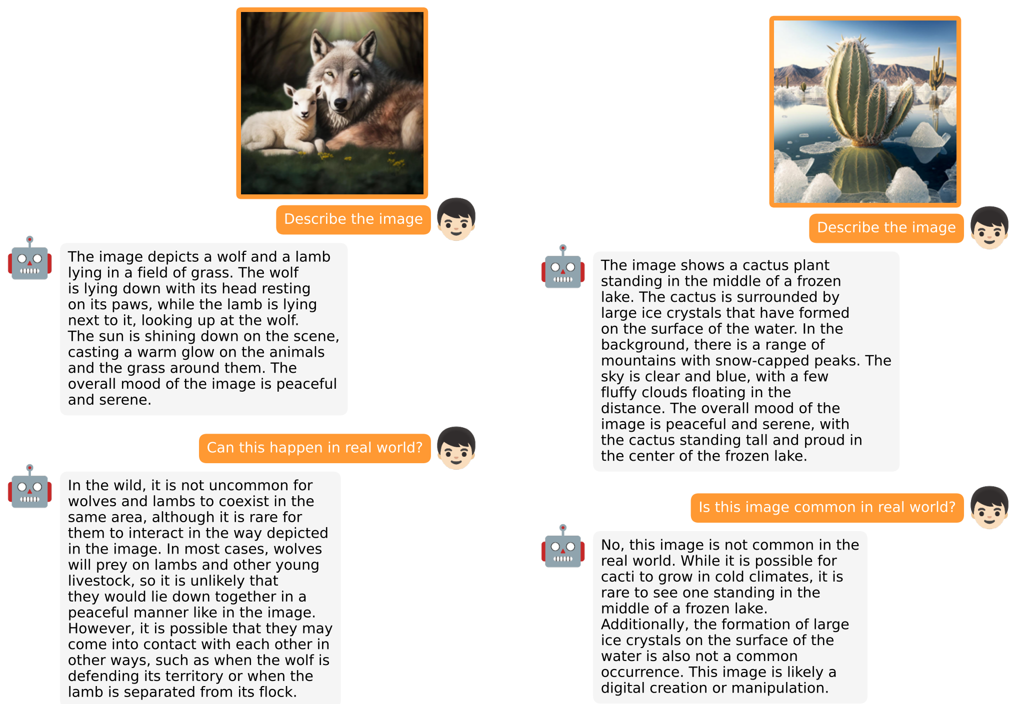
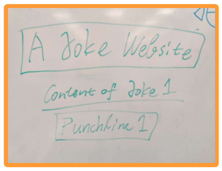


Figure 4: Discovering unusual content (Images are from WHOOPS dataset [3])



Write brief HTML/JS to turn this mock-up into a colorful website, where the jokes are replaced by a real joke

HTML:
```php
<!DOCTYPE html>
<html>
<head>
<title>A Joke Website</title>
<link rel="stylesheet"
type="text/css" href="style.css">
</head>
<body>
<div id="header">
<h1>A Joke Website</h1>
<nav>
<ul>
<li><a href="#">Home</a></li>
<li><a href="#">About</a></li>
<li><a href="#">Contact</a></li>
</ul>
</nav>
</div>
<div id="main">
<h2>Today's Joke</h2>
<p>Why did the tomato turn red?</p>
<p>Because it saw the salad
dressing!</p>
</div>
<div id="footer">
<p>&copy; 2023 A Joke Website</p>
</div>
<script src="script.js"></script>
</body>
</html>
```
CSS:
```css
body {
background-color: #f2f2f2;
font-family: Arial, sans-serif;
}

#header {
background-color: #333;
color: #fff;
padding: 20px;
text-align: center;
}

#header h1 {
margin: 0;
}

#header nav ul {
list-style: none;
margin: 0;
padding: 0;
}

#header nav ul li {
display: inline-block;
margin-right: 20px;
}

#header nav a {
color: #fff;
text-decoration: none;
}

#main {
background-color: #fff;
padding: 20px;
text-align: center;
}

#main h2 {
margin: 0;
}

#main p {
margin: 20px 0;
}

#footer {
background-color: #333;
color: #fff;
padding: 20px;
text-align: center;
}
```
JS:
```javascript
// Today's Joke
var joke = "Why did the tomato turn
red? Because it saw the salad
dressing!";

document.getElementById("main").innerHTML
= joke;
```

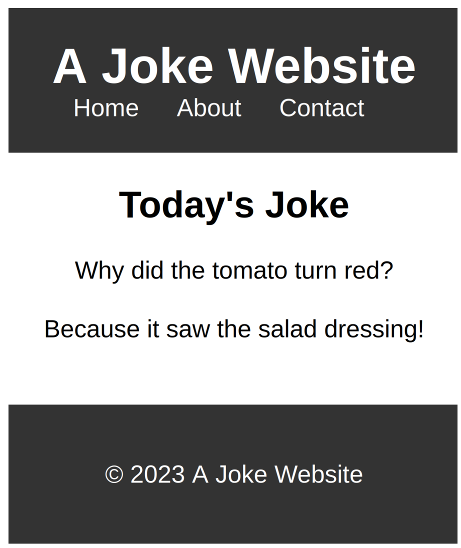


Figure 5: Generating website code from handwritten text and the rendered website

What is the issue with my plant? What should I do now?

The image shows a leaf with brown spots on it. The spots are likely caused by a fungal infection, which can be treated with a fungicide. It's important to identify the type of fungus causing the spots in order to choose the appropriate treatment.

To treat the plant, you can try the following steps:

1. Identify the type of fungus causing the spots on the leaves.
2. Use a fungicide specifically labeled for the type of fungus you have identified.
3. Apply the fungicide according to the product's instructions, making sure to cover all affected leaves.
4. Repeat the treatment every 7 to 14 days, or as directed by the product's instructions, until the fungus is completely gone.
5. Water the plant well after treatment to help the fungicide penetrate the leaves.
6. Keep the plant in a well-ventilated area and away from direct sunlight to prevent overheating.
7. Monitor the plant's progress and repeat the treatment if new spots appear.

It's also important to keep the plant healthy by providing it with proper care, such as adequate watering, fertilization, and pruning. This will help the plant to recover and prevent future fungal infections.

Why this happens and how to fix it?

The image shows a washing machine with a large amount of bubbles coming out of it. It appears that the washing machine is overflowing with soap suds.

There are several reasons why this might happen:

1. Overloading the washing machine: If too much laundry is put into the washing machine at once, it can cause an overflow.
2. Using too much detergent: If too much detergent is used in the washing machine, it can cause an overflow.
3. A clogged drain: If the drain of the washing machine is clogged, it can cause an overflow.
4. A malfunctioning washing machine: If the washing machine is not functioning properly, it can cause an overflow.

To fix this issue, you can try the following:

1. Check the load size: Make sure you are not overloading the washing machine.
2. Use the right amount of detergent: Use the recommended amount of detergent for the size of the load.
3. Clear the drain: Make sure the drain is clear of any debris.
4. Check for malfunctions: Have the washing machine checked by a professional if it is not functioning properly.

Figure 6: Identifying problems from photos and providing solutions

Can you write me a beautiful poem about this image?

Sure, here is a poem about the image you provided:

A man sits on a hill, with his dog by his side
The sun sets behind them, casting a golden light
The city below is bustling, but they are at peace
Enjoying the moment, in each other's embrace

The man's arms are wrapped around his furry friend
As they watch the world go by, with a gentle bend
The dog's head rests on the man's shoulder
A symbol of their unbreakable bond, forever

The sky is ablaze with color, a beautiful sight
But it's the love between the man and his dog that shines bright
It's a moment of pure joy, a memory to keep
Forever etched in their hearts, as they sit and weep

So let us cherish these moments, with our four-legged friends
For they bring us so much happiness, until the very end
And even when they're gone, their love will live on
In the memories we hold, forever and strong.

Can you write me a master rap song that rhymes very well based on this image?

Sure, here's a rap song based on the image you provided:

Verse 1:
I'm an old lady, but I'm still flyin'
I've got my headphones on and I'm feelin' high
I'm in the studio, makin' some beats
I'm the queen of hip hop, can't nobody touch my feet

Chorus:
I'm the grandma of rap, I'm still spittin' fire
I've got my green jacket on, I'm lookin' fly
I've got my sunglasses on, I'm shining bright
I'm the queen of the game, I'm makin' it right

Verse 2:
I've been in the game for years, I've paid my dues
I've got a microphone in my hand and I'm singin' the blues
I've got the crowd in the palm of my hand
I'm the queen of rap, I'm takin' over the land

Chorus:
I'm the grandma of rap, I'm still spittin' fire
I've got my green jacket on, I'm lookin' fly
I've got my sunglasses on, I'm shining bright
I'm the queen of the game, I'm makin' it right

Verse 3:
I've got my beats in my headphones, I'm feelin' the vibe
I'm in the zone, I'm in my element, I'm in my ride
I'm the queen of rap, I'm in control
I'm makin' hits, I'm takin' over the world

Chorus:
I'm the grandma of rap, I'm still spittin' fire
I've got my green jacket on, I'm lookin' fly
I've got my sunglasses on, I'm shining bright
I'm the queen of the game, I'm makin' it right

Outro:
I'm the grandma of rap, I'm still spittin' fire
I've got my green jacket on, I'm lookin' fly
I've got my sunglasses on, I'm shining bright
I'm the queen of the game, I'm makin' it right.

Figure 7: Rhyme generation

Can you write a story around this image for my child?

Sure, here's a story based on the image:

Once upon a time, there was a brown bear, a white rabbit, and a black cat who lived in a forest. They were good friends and loved to spend time together. One day, they decided to have a picnic under a tree. The bear brought some honey, the rabbit brought some carrots, and the cat brought some milk.

As they sat around the table, the bear said, "I'm so hungry, I could eat a whole honey pot!" The rabbit giggled and said, "I'm so hungry, I could eat a whole bag of carrots!" The cat meowed and said, "I'm so hungry, I could drink a whole bowl of milk!"

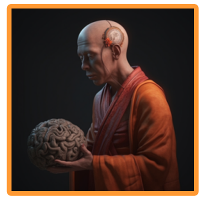
They all laughed and dug in to their food. The bear loved the honey, the rabbit loved the carrots, and the cat loved the milk. They were all very happy and content.

After they finished their meal, they decided to go for a walk in the forest. They came across a stream and the bear said, "Let's go for a swim!" The rabbit and cat were a bit hesitant, but the bear convinced them to join in.

They all had a great time swimming and playing in the water. The bear loved to splash around, the rabbit loved to dive, and the cat loved to chase the splashes. They were all very happy and had a lot of fun.

After their swim, they decided to go back to their picnic spot and have some more food. They sat around the table and the bear said, "I'm so hungry, I could eat a whole honey pot again!" The rabbit giggled and said, "I'm so hungry, I could eat a whole bag of carrots again!" The cat meowed and said, "I'm so hungry, I could drink a whole bowl of milk again!"

They all laughed and dug in to their food. They were all very happy and content. And they lived happily ever after.



I designed a character for my game as shown in the image. Can you write me a background story for him?

Sure, here's a possible background story for the character in the image:

The monk in the image is a wise and enlightened being who has spent many years studying the workings of the mind. He has spent countless hours meditating and contemplating the nature of consciousness, and as a result, he has gained a deep understanding of the inner workings of the mind.

The brain in his hand is a symbol of his deep understanding of the mind. He uses it to help others understand the nature of their own thoughts and emotions, and to guide them on the path to enlightenment.

Despite his great wisdom, the monk is a humble and compassionate being. He uses his knowledge to help others, and is always willing to share his insights with those who seek his guidance. He is a revered figure in his community, and many come to him seeking his help and guidance.

The monk's robes are a symbol of his commitment to his spiritual path. He wears them as a reminder of his vows of poverty, chastity, and obedience, and as a symbol of his commitment to living a life of service to others.

Overall, the monk in the image is a wise and compassionate being who has dedicated his life to understanding the nature of the mind and helping others on the path to enlightenment.

Figure 8: Story generation

We design new mugs shown in the image. Can you help us write an advertisement?

Sure, here's an advertisement for the mugs:

Introducing our new line of mugs featuring adorable cat designs! These mugs are perfect for coffee lovers and cat enthusiasts alike. Made from high-quality ceramic, each mug is durable and dishwasher safe. The unique designs are hand-painted, making each mug a one-of-a-kind piece of art.

Whether you're sipping your morning coffee or enjoying a warm drink on a chilly evening, these mugs are the perfect way to add a touch of feline charm to your daily routine. They also make great gifts for cat lovers in your life.

So why settle for a plain old mug when you can have a cat-themed one? Order yours today and start enjoying your beverages in style!

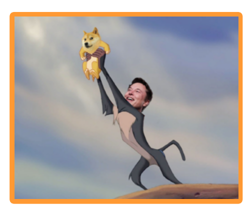
Help me write an advertisement for this.

Are you looking for a unique and stylish lighting solution for your home or office? Look no further than our brass toucan lamp! This beautifully crafted bird lamp is sure to add a touch of whimsy and charm to any room. The toucan's beak is designed to hold a light bulb, providing a warm and inviting glow. The base is made of wood, adding a natural touch to the overall design. The toucan lamp is a great addition to any room, whether you're looking to brighten up a dark corner or add a touch of personality to your space. Order yours today and bring a touch of the tropics to your home or office.

Figure 9: Product advertisements

Who are the people in the image?

The people in the image are Goku and Vegeta, two characters from the Dragon Ball series. Goku is a saiyan warrior and the main protagonist of the series, while Vegeta is a saiyan prince and the main antagonist. They are both known for their powerful abilities and their rivalry.



Who is the person in the image?

The person in the image is Elon Musk, the CEO of Tesla and SpaceX. He is known for his work in the fields of electric cars, space exploration, and renewable energy. He is also known for his eccentric personality and outspoken views on various topics.

Figure 10: Individuals Identification

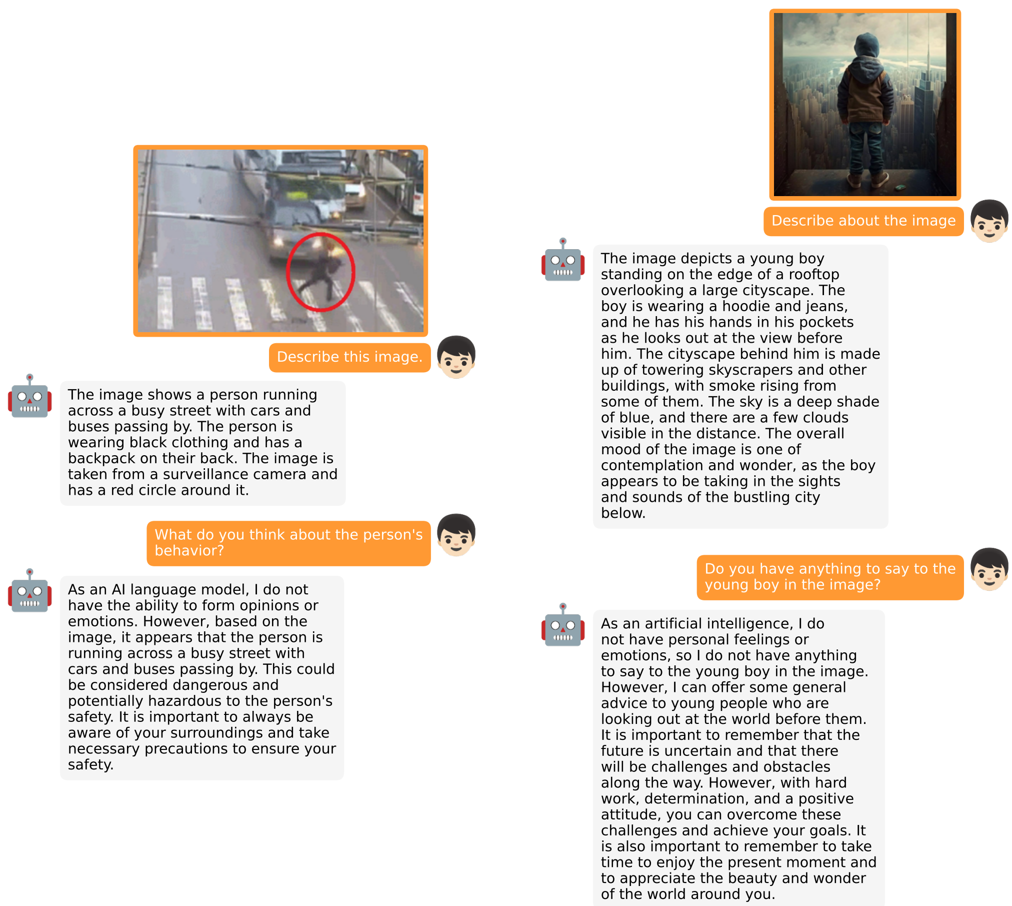


Figure 11: Image commenting

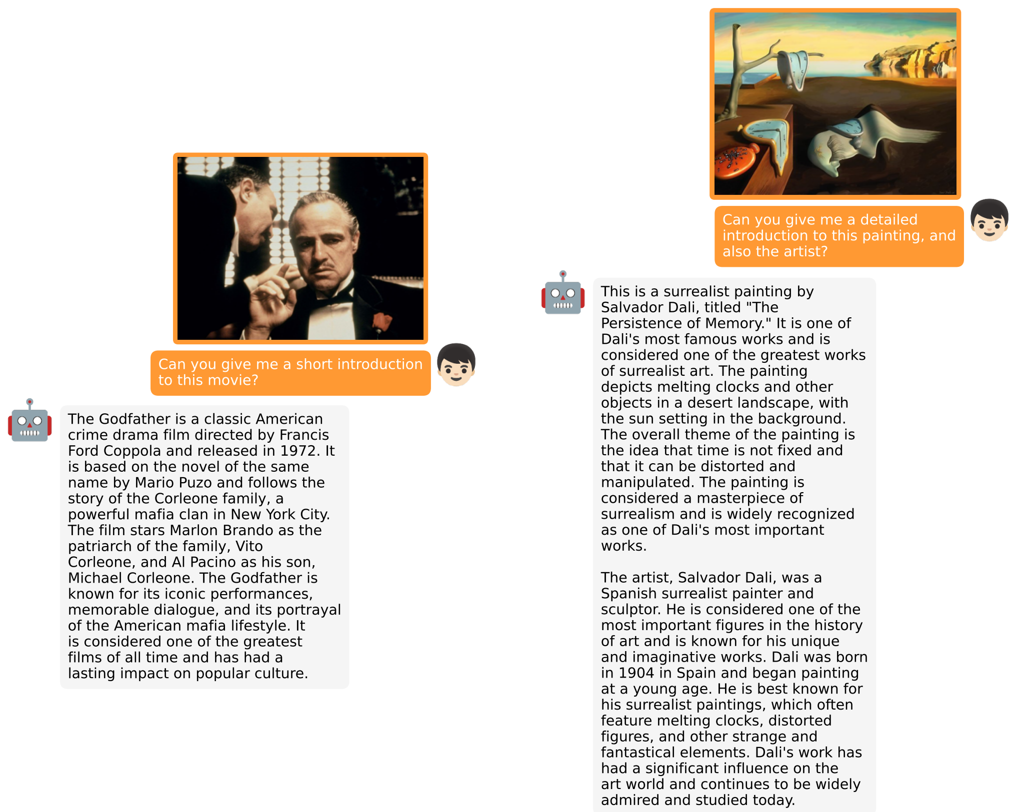
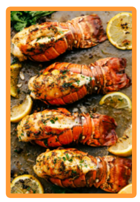


Figure 12: Fact retrieval



How should I make something like this?

To make this dish, you will need the following ingredients:

* 4 large lobsters
* 2 tablespoons olive oil
* 2 cloves garlic, minced
* 1 teaspoon dried thyme
* 1 teaspoon dried oregano
* 1/2 teaspoon paprika
* Salt and pepper, to taste
* 4 lemon wedges, for serving

1. Preheat the oven to 400°F (200°C).
2. In a large baking dish, toss the lobsters with the olive oil, garlic, thyme, oregano, paprika, salt, and pepper.
3. Roast the lobsters in the preheated oven for 20-25 minutes, or until they are cooked through.
4. Remove the lobsters from the oven and let them cool for a few minutes.
5. Serve the lobsters with the lemon wedges on the side.

I hope this helps! Let me know if you have any other questions.

How should I do to cook this?

To cook ramen noodles, you will need the following ingredients:

1. Ramen noodles
2. Pork or chicken broth
3. Soy sauce
4. Sesame oil
5. Green onions
6. Sesame seeds
7. Sliced pork or chicken
8. Boiled eggs

Here's how to cook ramen noodles:

1. Bring the pork or chicken broth to a boil in a large pot.
2. Add the soy sauce, sesame oil, and green onions to the broth.
3. Add the ramen noodles to the broth and cook according to the package instructions.
4. While the noodles are cooking, cook the sliced pork or chicken in a separate pan.
5. Once the noodles are cooked, drain them and add them to the pan with the pork or chicken.
6. Add the boiled eggs to the pan and stir everything together.
7. Serve the ramen noodles in bowls, topped with sesame seeds and green onions.

I hope this helps! Let me know if you have any other questions.

Figure 13: Food recipes generation

References

[1] Sharegpt. `https://github.com/domeccleston/sharegpt`, 2023.

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

[3] Nitzan Bitton-Guetta, Yonatan Bitton, Jack Hessel, Ludwig Schmidt, Yuval Elovici, Gabriel Stanovsky, and Roy Schwartz. Breaking common sense: Whoops! a vision-and-language benchmark of synthetic and compositional images. *arXiv preprint arXiv:2303.07274*, 2023.

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

[5] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. 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, 2021.

[6] Jun Chen, Han Guo, Kai Yi, Boyang Li, and Mohamed Elhoseiny. Visualgpt: Data-efficient adaptation of pretrained language models for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18030–18040, 2022.

[7] Jun Chen, Deyao Zhu, Kilichbek Haydarov, Xiang Li, and Mohamed Elhoseiny. Video chatcaptioner: Towards the enriched spatiotemporal descriptions. *arXiv preprint arXiv:2304.04227*, 2023.

[8] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023.

[9] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

[10] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022.

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

[12] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

[13] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. *arXiv preprint arXiv:2211.07636*, 2022.

[14] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

[15] Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, et al. Language is not all you need: Aligning perception with language models. *arXiv preprint arXiv:2302.14045*, 2023.

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

[17] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022.

[18] OpenAI. Introducing chatgpt. *https://openai.com/blog/chatgpt*, 2022.

[19] OpenAI. Gpt-4 technical report, 2023.

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

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

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

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

[24] Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022.

[25] Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022.

[26] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400

million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

[27] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 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, 2018.

[28] Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, et al. Using deepspeed and megatron to train megatron-turing nlg 530b, a large-scale generative language model. *arXiv preprint arXiv:2201.11990*, 2022.

[29] Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. *arXiv preprint arXiv:2303.08128*, 2023.

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

[31] Anthony Meng Huat Tiong, Junnan Li, Boyang Li, Silvio Savarese, and Steven CH Hoi. Plug-and-play vqa: Zero-shot vqa by conjoining large pretrained models with zero training. *arXiv preprint arXiv:2210.08773*, 2022.

[32] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[33] Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34:200–212, 2021.

[34] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022. Survey Certification.

[35] Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*, 2023.

[36] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Zero-shot video question answering via frozen bidirectional language models. *arXiv preprint arXiv:2206.08155*, 2022.

[37] Zhengyuan Yang*, Linjie Li*, Jianfeng Wang*, Kevin Lin*, Ehsan Azarnasab*, Faisal Ahmed*, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. Mm-react: Prompting chatgpt for multimodal reasoning and action. 2023.

[38] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

[39] Deyao Zhu, Jun Chen, Kilichbek Haydarov, Xiaoqian Shen, Wenxuan Zhang, and Mohamed Elhoseiny. Chatgpt asks, blip-2 answers: Automatic questioning towards enriched visual descriptions. *arXiv preprint arXiv:2303.06594*, 2023.