BLIP Vision-Language Model Integration

A Python implementation for integrating the BLIP (Bootstrapping Language-Image Pre-training) model for visual question answering.

1. Prerequisites

Before running the code, you need to install the necessary dependencies. Here's how to set up your environment:

Run this command to install all required packages:

```
pip install -r requirements.txt
```

Make sure you have Python 3.10+ installed, along with PyTorch with CUDA support if you're using a GPU.

2. Usage

2.1 Initialize the BLIP Model (Singleton Class)

To ensure that the model is loaded only once, we use a singleton pattern for the Blip class. Here's a breakdown of how this is done.

Singleton Pattern: The Blip class only initializes once and uses that instance for subsequent calls.

```
class Blip:
    _instance = None
    _initialized = False

    def __new__(cls):
        if cls._instance is None:
            cls._instance = super(Blip, cls).__new__(cls)
        return cls._instance

    def __init__(self):
        if not Blip._initialized:
            self.processor = AutoProcessor.from_pretrained("Salesforce/blip-vqa-base")
            self.model = BlipForQuestionAnswering.from_pretrained("Salesforce/blip-vqa-base'
            Blip._initialized = True
```

- Processor: Converts the image and question into input tensors for the model.
- Model: Loads the BLIP model and moves it to the GPU (cuda).
- Singleton: Ensures that the model and processor are initialized only once.

3. Test Function to Ask Questions

This function takes an image path and a list of questions. It processes the image and answers each question using the BLIP model.

Function Explanation:

```
def test_blip_with_questions(blip=None, image_path="image1.webp", questions=None):
    if blip is None:
        blip = Blip()  # Initialize the singleton if not already done

    try:
        image = Image.open(image_path)  # Open the image
    except Exception as e:
        print(f"Error opening image {image_path}: {e}")
        return

    for q in questions:
        try:
            print(f"Q: {q}\nA: {blip.ask(image, q)}")  # Ask the question and print the answ
        except Exception as e:
            print(f"Error processing question '{q}' for image {image_path}: {e}")
```

- Load Image: The image is opened using the PIL library.
- Ask Questions: The questions are passed to the model, and answers are generated.

4. How to Ask a Question

To ask a question about an image, you can use the `ask` method from the `Blip` class. The method takes an image and a question as input and returns an answer.

Example:

Initialize the Blip Model: You can initialize the model as follows:

```
blip = Blip()  # Singleton instance
```

Ask a Question: You can call the ask method with an image and a question:

```
answer = blip.ask(image, "What is in the image?")
print(answer)
```

- image: This is the image you want to ask questions about. It should be opened using PIL (Python Imaging Library).
- “What is in the image?”: This is the question you are asking about the image. You can replace this with any other valid question.

Full Example: Here is a complete example to demonstrate how to ask questions about an image:

```
from PIL import Image
from blip import Blip

# Load your image
image = Image.open("path_to_your_image.jpg")

# Initialize the BLIP model
blip = Blip()

# Ask questions
question = "What is in the image?"
answer = blip.ask(image, question)

# Print the answer
print(f"Q: {question}\nA: {answer}")
```

Example Output:

```
Q: What is in the image?
A: A cat sitting on a chair.
```

5. Command-Line Arguments for Flexibility

To run the script with custom parameters (like the image path and questions), we use the argparse module to handle command-line arguments.

Here's how you set up the argument parsing:

```
if __name__ == "__main__":
    parser = argparse.ArgumentParser(description="Test BLIP with questions on an image.")
    parser.add_argument("--image_path", type=str, default="image1.webp", help="Path to the 
    parser.add_argument("--questions", nargs='+',
                        default=["What are those in the image?",
                                "How many cars are there?",
                                "Is there a blue car in the image?"],
                        help="List of questions for the image.")

    args = parser.parse_args()
    test_blip_with_questions(image_path=args.image_path, questions=args.questions)
```

6. Running the Script

To run the script, execute the following command in your terminal:

```
python test_blip.py --image_path path_to_your_image.jpg --questions "What is in the image?"
```

This will process the image located at `path_to_your_image.jpg` and ask the provided questions.

7. Troubleshooting

- CUDA issues: If you encounter any errors related to GPU/CPU, ensure that your CUDA environment is set up correctly.
- Image loading: Double-check the file path to ensure the image exists and is accessible.
- Dependencies: Ensure all dependencies are installed by running pip install -r requirements.txt.