Lab File for Agentic AI

Sharda School of Engineering and Technology

Name- Divek Jain
Sys ID- 2023203980
Semester/Year- 6th, 3rd Year
Class: CS-H G2

Faculty-In-Charge/Submitted To
Mr.Ayush Kumar Singh

Sharda University

Plot No. 32-34,
Knowledge Park III,
Greater Noida, Uttar
Pradesh 201310

Fine-Tuning BLIP for Image Captioning Documentation

1. Introduction

This project focuses on fine-tuning the **BLIP (Bootstrapped Language-Image Pretraining)** model for the task of **image captioning**. Image captioning is a multimodal problem where a system generates meaningful textual descriptions for given images. Instead of training a model from scratch, this project leverages a pre-trained BLIP model and adapts it to a custom image–caption dataset using fine-tuning techniques.

---

2. Objectives

- To understand how vision–language models work.
- To fine-tune a pre-trained BLIP model on a custom dataset.
- To improve caption quality for domain-specific images.
- To evaluate model performance before and after fine-tuning.

---

3. Technologies and Tools Used

- **Python** – Core programming language
- **PyTorch** – Deep learning framework
- **Hugging Face Transformers** – Pre-trained BLIP model and utilities
- **BLIP Processor** – For image and text preprocessing
- **Jupyter Notebook** – Experimentation and training

---

4. Dataset Description

The dataset consists of paired samples of: - **Images** (input) - **Captions** (ground truth text descriptions)

Each data point teaches the model how a particular image should be described in natural language.

Example: - Image: image_01.jpg - Caption: “A dog playing with a ball in a park”

---

5. Model Overview

5.1 BLIP Architecture

BLIP is a vision–language model composed of: - **Image Encoder** – Extracts visual features from images - **Text Decoder** – Generates captions based on visual features

The encoder processes the image, while the decoder predicts the caption token by token.

---

6. Fine-Tuning Methodology

6.1 Loading the Pre-trained Model

A pre-trained BLIP model is loaded to take advantage of existing visual and linguistic knowledge. This significantly reduces training time and data requirements.

6.2 Data Preprocessing

- Images are converted into pixel tensors.
- Captions are tokenized into numerical IDs.
- Both are handled using the BLIP Processor.

6.3 Training Process

For each image–caption pair: 1. The image is passed through the image encoder. 2. The model predicts a caption. 3. A loss value is computed by comparing the predicted caption with the actual caption. 4. The model weights are updated using backpropagation.

This process is repeated over multiple epochs to improve performance.

---

7. Loss Function and Optimization

- **Loss Function**: Cross-entropy loss for text generation
- **Optimizer**: AdamW

The loss indicates how far the predicted caption deviates from the ground truth. Lower loss implies better caption accuracy.

---

8. Evaluation

After fine-tuning: - The model is tested on unseen images. - Generated captions are compared with expected descriptions. - Improvements are observed in contextual accuracy and relevance.

---

9. Results

- More descriptive and domain-relevant captions
- Reduced generic outputs
- Better alignment between image content and generated text

---

10. Applications

- Automated image description systems
- Assistive technologies for visually impaired users
- Content moderation and image indexing
- Social media and e-commerce image tagging

---

11. Limitations

- Performance depends on dataset quality
- Requires GPU for efficient training
- Overfitting may occur with very small datasets

---

12. Conclusion

This project demonstrates how a powerful pre-trained vision–language model like BLIP can be effectively fine-tuned for image captioning tasks. Fine-tuning enables the model to adapt to specific domains while retaining its general understanding of images and language.

---

13. Future Scope

- Training on larger and more diverse datasets
- Experimenting with different vision–language models
- Deploying the model as a web or mobile application