Final Project: The Visual Storyteller Overview

Dataset: caption_data.zip

In this assignment, you will build a system that performs image captioning. Your goal is to construct a model that takes an image as input and produces a natural language description as output. This task requires your system to bridge two distinct modalities: it must interpret visual information and express that understanding through generated text.

The Challenge

The core challenge is translation across domains. Unlike standard translation (text-to-text), this task starts with unstructured pixel data. Your model must learn to identify visual content—objects, actions, and settings—and map them to correct linguistic structures.

Furthermore, this is an open-ended generation task. A single image can be correctly described in many different ways (e.g., "A dog on the grass" vs. "An animal playing outside"). Your system must learn to generate captions that are not just grammatically correct, but semantically faithful to the visual input.

Technical Requirements

1. The Data

You will be working with the **provided** dataset.

- **Composition:** The dataset consists of 8,000 unique images.
- **Annotations:** Each image is paired with five different captions written by humans.
- **Nature of Data:** You are provided with the raw image files and the corresponding text files containing the descriptions.

2. The Model

You must design and implement a neural network capable of sequence generation conditioned on visual input.

- **Input:** The model must accept an image.
- **Output:** The model must generate a sequence of words (a sentence) describing that image.

Deliverables

Notebook 1: **data_and_training.ipynb**
This notebook serves as the documentation of your development process. It must include:

1. **Data Loading:** Mechanisms to ingest the images and text.
2. **Model Definition:** The code defining your network architecture.

3. **Training:** The execution of your training process, showing the progression of the model's learning (e.g., loss metrics).
4. **Save:** The mechanism to save your final trained model artifacts.

Notebook 2: **inference.ipynb**

This notebook demonstrates your trained model in a production-like setting. It must contain:

```
def generate_caption(image_path: str, model: any) -> str:
    """
    Takes a path to an image and returns a generated caption string.
    """
    pass
```

- **Demonstration:** Application of your model on a set of unseen test images.
- **Analysis:** Examples of both successful captions and failure cases.

Submission Format

- **Group Projects:** A **GitHub repository link is mandatory**. The repository must contain the notebooks, a README, and instructions on how to run the code. ***Do not*** write code offline and upload with 1 commit, I want to see that more or less everybody contributed equally.
- **Individual Projects:** A GitHub link is **suggested**, but a ZIP file upload containing the notebooks and model artifacts is accepted.

Due Date

23 January 23:59
