**Image Classification using CLIP**

**📝 Overview**

This project utilizes **OpenAI's CLIP (Contrastive Language-Image Pretraining)** for zero-shot image classification. CLIP is a **powerful multimodal model** that can understand images and text in the same representation space, allowing it to classify images **without task-specific training**.

**🚀 How It Works**

1. **Load the CLIP Model** – The pre-trained CLIP model is used to extract embeddings from both images and text labels.
2. **Process the Input Image** – The image is transformed into a feature vector using CLIP’s vision encoder.
3. **Process the Text Labels** – Text descriptions of possible classes are encoded into vectors using CLIP’s text encoder.
4. **Compute Similarity** – The model calculates the similarity between the image features and text embeddings.
5. **Predict the Best Match** – The label with the highest similarity score is chosen as the classification result.

**🖼 Example**

If given an image of a **cat**, and text labels like:  
✅ "a cat"  
✅ "a dog"  
✅ "a bird"  
CLIP will identify the closest match and classify the image as **"a cat"** based on similarity scores.

**📦 Dependencies**

* Python 3.x
* OpenAI CLIP (pip install open\_clip\_torch)
* Torch & torchvision

**🏁 Getting Started**

1. Clone this repository:

git clone https://github.com/your-username/clip-image-classification.git

cd clip-image-classification

1. Install dependencies:

pip install -r requirements.txt

1. Run the classification script:

python classify.py --image path/to/image.jpg

**📖 References**

* [OpenAI CLIP Paper](https://arxiv.org/abs/2103.00020)
* [CLIP GitHub Repository](https://github.com/openai/CLIP)