A GeoGuessr Bot

This document summarizes my goal for the upcoming final project and describes many details of the project, such as the datasets being used, training, and evaluation metrics. I'm working on this project alone.

Summary

This model will be able to play the game GeoGuessr, which revolves around trying to guess the exact location of a google street view image. This bot will move around a round, grab 3 images of a round, and attempt to guess where in the US an image was taken. An accurate model (80% correct country classification) could have a positive impact on climate research, as if an AI could figure out that two images came from the same region, you would be able to study how climate change affected that area. There are also a lot of other small ways that this type of model would be helpful to environmental researchers and others, if it were extended to the rest of the world. This extension could be done fairly easily, but is out of the scope of this project due to the extra cost required to gather street view images from around the world.

Project Overview and Approach

The dataset will be gathered by using the street view static api. We will split the US into approximately 150 equally sized squares, randomly generate 60 valid locations within each square, and then take 3 different photos (with different angle rotations) for each of those locations. This will result in approximately 27,000 images for our model to train on. For each set of images in the dataset, we will get the model prediction, and compute how far away the prediction is from the actual location, and determine the loss (scaled-down MSE). There will be a 90/10 split for training and validation. The goal is for our prediction to be less than 50 miles away for any US location that we put into the model.

Responsibilities:

Anuraag - Split US into 150 squares, get training data based on the street view api, design the model architecture, train on a smaller subset of data to test the model parameters and layers, train the whole dataset, and then run an evaluation routine on the model. Multiple trainings might be necessary in order to get a model that operates within the 50 mile error limit

As a bonus, I want this model to be able to play geoguessr directly, so a bot needs to be built that takes pictures in the browser and connects with the model.