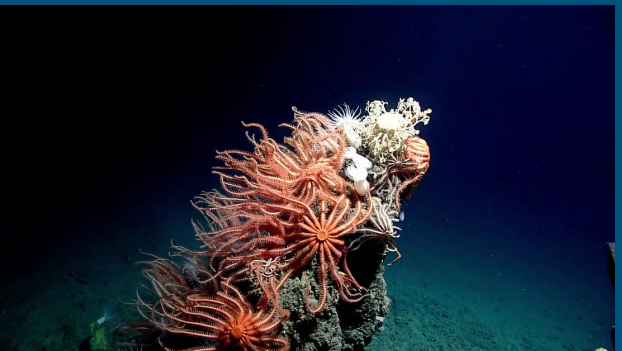
Deep Sea Image Classification



Cloud Computing

Jennifer Crosby
Benjamin Lee
Polly McKim

Overall Project

- Acquired deep sea research expedition images from NOAA
- Built and deployed a machine learning model to classify images as interesting or not interesting based on creature presence
- Used AWS architecture to greatly increase speed and scalability of the model
- Delivered a model of close to 70% accuracy
- Used a CNN (convolutional neural network) algorithm as our model

Sources Used

- NOAA repository of deep sea images
- CVision AI (http://cvisionai.com/) houses NOAA images on an online server
- 1500 images + additional created through image augmentation

Features Implemented

Python libraries to install:

Pandas, numpy , matplotlib, sklearn, pytorch, pytorchvision,

warnings, PIL, image_folder, random

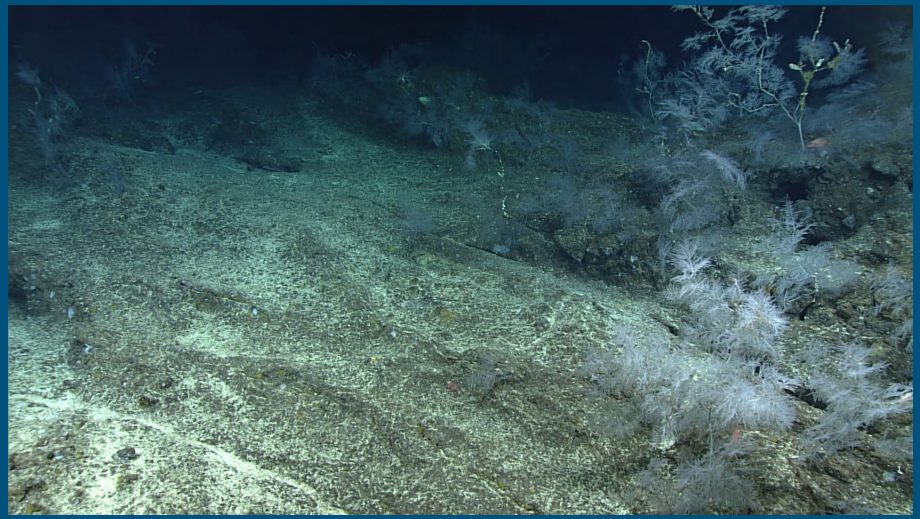
To use in model:

CNN, Adam optimizer, back propagation

JKC

Examples of Inputs

Not Interesting

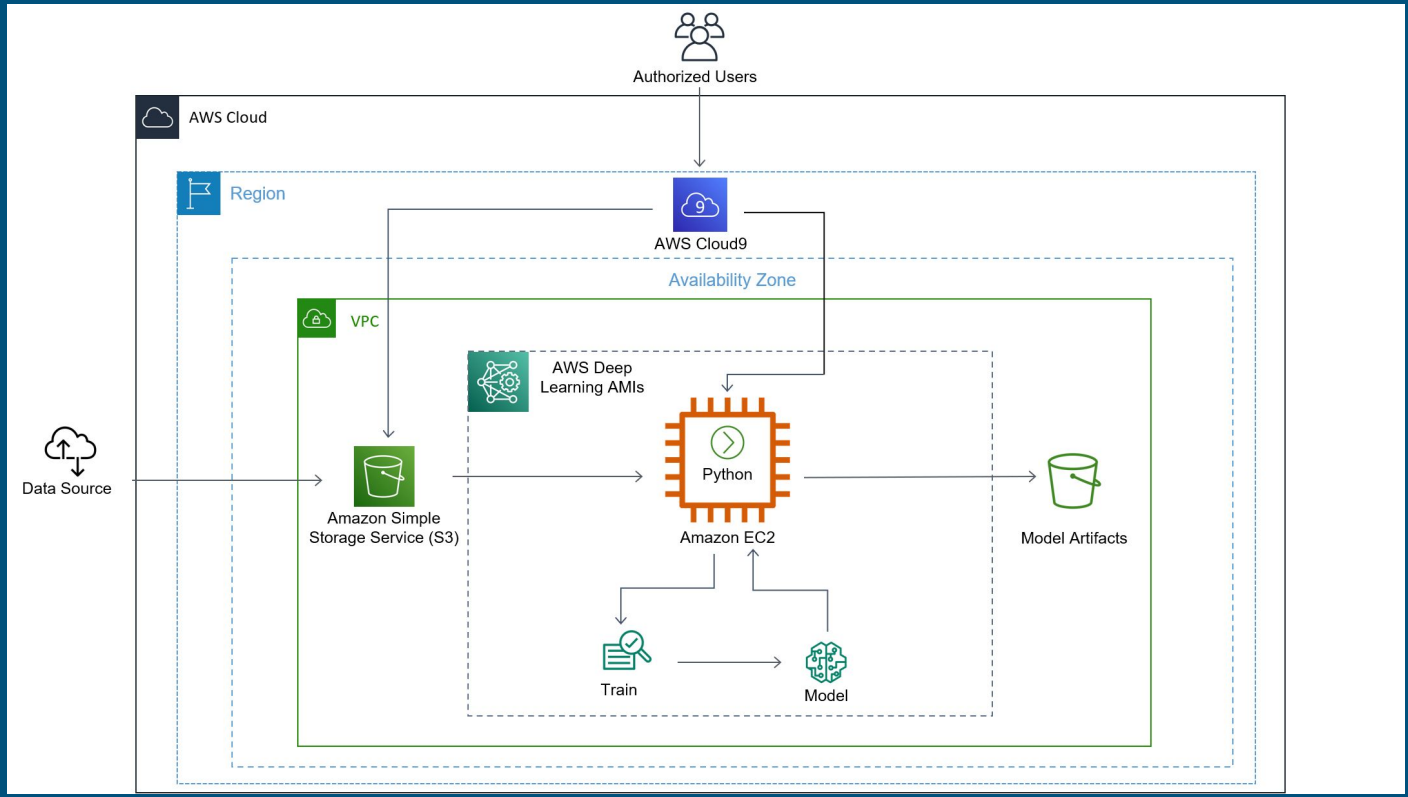


Interesting

AWS Architecture Used for Project

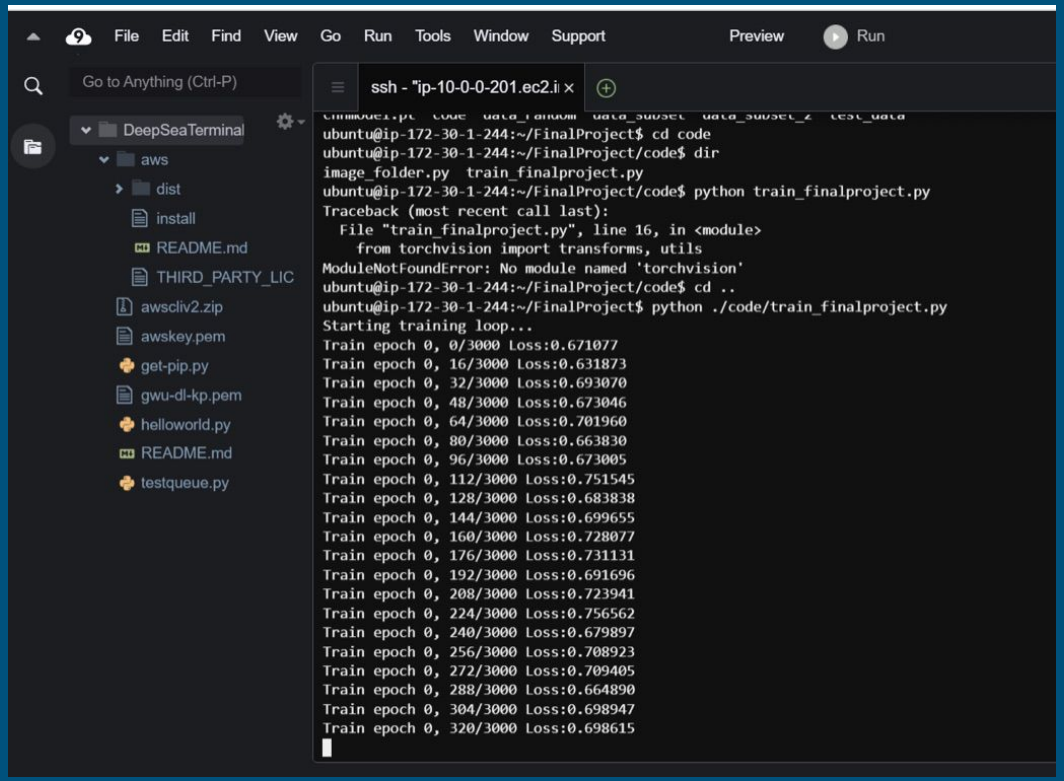
- EC2 (AMI Name Deep Learning Base AMI (Ubuntu 18.04) Version 31.0); Type: g3.4xlarge
- VPC
- AWS Cloud9- Cloud IDE

Architecture for Deep Learning EC2 Deployment



Cloud Architecture Demo

Python Running the Model on the Deep Learning EC2



68%

Overall Accuracy Achieved

Findings & Cautionary Tales

- Ran much faster than on a typical personal computer
- Auto Scalability or using more GPUs would likely have improved the run time even more but would have added to the cost of running the model
- Understanding of EC2 Ubuntu choices for deep learning is important (ie make sure the family you select has the capacity and features you need)
- GPUs do not exist in all availability zones or on every option of deep learning Ubuntu (Don’t select a T when you really need a G or a P!)
- The future the process would be simplified by figuring out feed the data directly from the S3 to the running python program on the EC2

Questions?