Annotation Assignment 1 Instructions

In this assignment, we will use our tool to conduct Active Learning for annotating forest and non-forest regions in satellite images. The idea is to make corrections on the labels predicted by an initial model and let the model learn from the corrections made to refine the predictions. For specifics on how to conduct the annotation, please follow the instructions given by the TAs during the demonstration.

To perform this assignment, we expect you to be able to use the Linux terminal and ssh into UAB's cheaha supercomputer. If you are a Windows user, please consider installing Windows Subsystem for Linux: https://learn.microsoft.com/en-us/windows/wsl/install so that you can use Linux terminal directly.

To use the tool in your local computer, please follow the instructions given below:

Step 1: Code Download
git clone https://github.com/saugatadhikari/forest_model_AL.git and put it in your desired folder in UAB's Cheaha server. Please only use the code in main branch.

Step 2: Dataset Download
1. Download the 2 zip files from the links below, extract them in the path below (where you have cloned the git repository): /forest_model_AL/backend_code/data_al/repo
   a. https://drive.google.com/file/d/1pQFycst8o1udDMTOkklQRkHCgl9J0Taa/view?usp=sharing
   b. https://drive.google.com/file/d/1mMX70XYqufMi-CYIh5BwCQnHSpKNq-i9/view?usp=sharing
2. Download the zip file from the link below and extract it in the path below: /forest_model_AL/: https://drive.google.com/file/d/1B1eeFZRY2jf6xzwuk40EH3E51UJZ-hVt/view?usp=sharing
3. Download the zip file from the link below and put them in a folder that's convenient to you in your local computer. You will need to upload the files present in this folder later to the frontend: https://drive.google.com/file/d/19yM2JogLxKcIS32bGLHUO-YJNNMoFunJ/view?usp=sharing

Step 3: Environment Setup
1. GPU env setup in Cheaha (one time thing)
   a. Open a terminal in your local computer
   b. Run this command: ssh BlazerId@cheaha.rc.uab.edu
   c. Enter your password (UAB's outlook account's password)
   d. cd to /forest_model_AL/
   e. Run this command: module load Anaconda3

f. open environment.yml file, replace 'BlazerId' with your BlazerId, and save the file
g. Run this command to install a new virtual env with all the requirements
    - conda env install -f environment.yml
h. Activate the installed environment using and make sure the environment is activated:
    - conda activate al_env
i. **Do not exit this terminal until you are done with the assignment.**

Step 4: Machine Allocation on Cheaha

From cheaha's terminal, run one of the following two commands. Try to use Amperenodes as first priority since it is faster than Pascalnodes.

1. **Amperenodes**:
    srun --ntasks=1 --cpus-per-task=1 --mem-per-cpu=32000 --time=12:00:00 --partition=amperenodes --job-name=JOB_NAME --gres=gpu:1 --pty /bin/bash

2. **Pascalnodes**:
    srun --ntasks=1 --cpus-per-task=1 --mem-per-cpu=32000 --time=12:00:00 –partition=pascalnodes --job-name=JOB_NAME --gres=gpu:1 --pty /bin/bash

Once this command is run, it will take some time to allocate you a machine. After the machine is allocated, you will see something like this in your terminal: **blazerid@c0xxx**. Please note down **c0xxx**, this is your machine id which will be used in Step 5 below.

Step 5: ssh into the allocated machine

Important: Please follow the instructions below in the exact order (otherwise it might not work):

Open 4 different terminals in your local computer and follow the instructions below. **Do not exit any of these terminals until you are done with the assignment.**

Terminal 1:
1. Login to cheaha: ssh BlazerId@cheaha.rc.uab.edu
2. ssh into allocated machine: ssh c0xxx (this is the id of machine allocated at Step 4
3. module load Anaconda3
4. conda activate al_env
5. cd into forest_model_AL/backend_code/data_al and run **python data_maker_al.py. Note: This step is only a one time thing!**

6. cd into forest_model_AL/backend_code/
7. Run flask app: flask --debug run --host=0.0.0.0 --port=5005

Terminal 2:
1. Login to cheaha: ssh BlazerId@cheaha.rc.uab.edu
2. ssh into allocated machine: ssh c0xxx (this is the id of machine allocated at Step 4
3. cd into forest_model_AL/src/client/
4. Run frontend app:
   a. export LD_LIBRARY_PATH=""
   b. npm run dev

Terminal 3:
`ssh —L 5005:c0xxx:5005` [`BlazerId@cheaha.rc.uab.edu`](mailto:BlazerId@cheaha.rc.uab.edu) from local machine's terminal
(remember to replace c0xxx with your machine id from Step 4)

Terminal 4:
`ssh —L 8082:c0xxx:8082` [`BlazerId@cheaha.rc.uab.edu`](mailto:BlazerId@cheaha.rc.uab.edu) from local machine's terminal
(remember to replace c0xxx with your machine id from Step 4)

Step 6: Running the application from local computer
- Open a browser (Google Chrome preferred)
- Navigate to chrome://settings/system and enable "**Use hardware acceleration when available**", without this the application does not work!
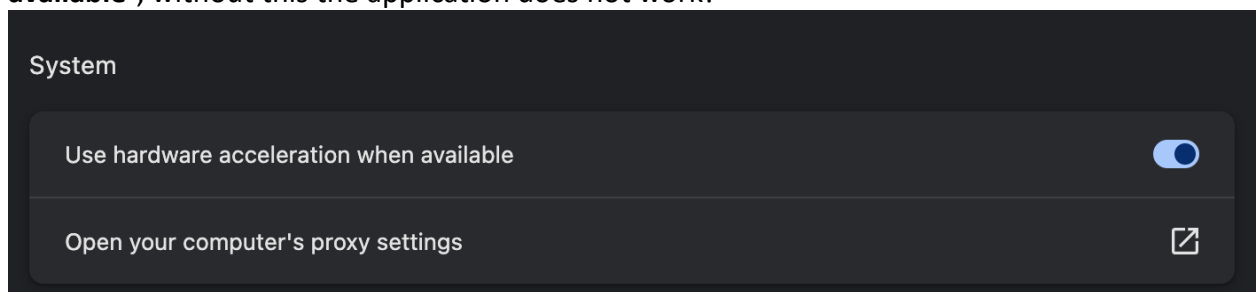
- Run localhost:8082 in the browser
- Enter your BlazerId in Student ID box, and test region id in the Test Region ID box
- Upload elevation png and RGB png for respective test regions and submit

What to Submit?

- A zip file with individual folders for all the test regions. Individual folders should be named Region_X_TEST (X is the region id). These folders each should contain 2 files below:
  - Final zip file that you downloaded from the frontend for this test region
  - Latest model checkpoint saved in /forest_model_AL/backend_code/users/YOUR_BLAZER_ID/saved_models_forest/Region_X_TEST/. Model checkpoints are named as saved_model_forest_1.ckpt, saved_model_forest_2.ckpt, saved_model_forest_3.ckpt…. and so on. You only need to submit the latest one i.e, saved_model_forest_3.ckpt in this case.

Known issue:

Frontend, sometimes, cannot call backend 127.0.0.1:5005 for some reason. If that happens:

1. go to Terminal 3 and type "exit" and Enter
2. go to Terminal 1 and hit Ctrl + C
3. run this command again on Terminal 1: flask --debug run --host=0.0.0.0 --port=5005
4. go to Terminal 3 and run this command: `ssh –L 5005:c0xxx:5005` BlazerId@cheaha.rc.uab.edu
5. Try to run the application on frontend and see if it hits the backend this time