2 Capstone Project: Tree Classification near Highways

2.1 Description

In this project, you are expected to collect data from different sources of web pages and develop deep neural networks for the task of tree classification, especially for trees located near highways. You are free to choose several tree types for the problem as well as the algorithm and the architecture you want to implement. This project is aimed to help decide when we need to do some maintenance (e.g. trim the trees due to their overgrowth in height or spread) or evaluate the risk of damaging nearby infrastructures caused by those trees, thus allowing us to send the onsite crews in a more efficient and effective manner.

2.2 Data Collection

This project is an open task and there is no pre-collected dataset for you to start with. You are supposed to define the problem and collect data from a variety of sources:

- Here is the link to the nurseries that you can search by Zones: *https://shop.baileynurseries.com/Product/Inventory?category=15-SHADE*
- In the link above, you will also need to parse detailed information for a given image, such as fast-growing, full-sun exposure, height, spread, zones, etc. Use the information wisely to determine not only the type of a tree but also the risk of its overgrowth.
- You may also need Google to obtain more labeled image data. For example, you can google "maple tree" or "maple tree highway" to get some test data to validate your model.

2.3 Potential Challenges

- Images coming from different resources might have different levels of details that require preprocessing
- In a highway image, there might be a bunch of trees instead of one in a standard tree image. So you will need to transfer the model trained on single-tree images.
- Computational cost training on image data
- No large dataset exists for you to pre-train the tree model

2.4 Submission Requirement

- Jupyter notebooks (ipynb) or python source code for training, inference, and evaluation of your model.
  - You may choose Tensorflow or PyTorch for implementation
  - implementation must be Python-based
- Instructions to set up and run the source code (Docker file, shell script, pip commands, etc.)
- PowerPoint slides and the corresponding PDF files illustrating the graph embedding method
- Any datasets you used or links to download the datasets
- Report with the results supporting your comparative analysis: include the following information and metrics at a minimum:
  - Description of the datasets
  - Description of the specific problem of your choice. (e.g. Which trees are you going to classify? What are the other objectives of your problem?)
  - Description of the specific algorithm
  - Architectures: visual graphs
  - Architectures Hyper-parameters: table
  - Time/Epoch: bar graphs
  - Hyper-parameter Tuning: Choices, rationale, observed impact on the model