Undergraduate AI Capstone NYCU Spr2022 Programming Assignment #1 Due 3/13/2022

The topic of this project is classification, a major part of supervised learning. It is likely that you have done projects on this topic in previous courses, and there are also a lot of material that you can find on the Internet. Here we are less concerned with particular algorithms, and more on the process, analysis, and presentation of the results. The requirements are listed below:

Datasets:

You need to use three datasets, one each from the following categories (note: Kaggle datasets and datasets from your other projects are allowed):

- ■ A public image dataset. There are a lot of them around. Here we want to focus on classification instead of detection, so sources like PASCAL VOC are not very suitable. Just Google "image classification datasets" and follow some links to find one for your interest.
- ■ A public non-image dataset. In most cases, this means that the data are already in the form of feature vectors. A good source is UCI Machine Learning Repository. Choose one intended for classification so that you have the class labels.
- ■ A self-made dataset. There are many possible options. The objective here is that you go through the process of data collection, labeling, and preprocessing yourself. Dataset size of a few dozens to a few hundreds is sufficient. Use your creativity. There are some notes:
    - Choose the topic such that (1) you can do labeling confidently, and (2) it does not require a huge amount of data to produce reasonable results.
    - For your chosen data type, try to use basic feature extractors available in public libraries, such as BoW for text, MLCC for sound, or LBP/HOG for images. Just go check them out. In many cases you can also build your own simple feature extractors. Exploring unfamiliar types of data will be more time consuming, but that will expand your knowledge as well.
    - It is allowed that a few of you collaborate to compose a single dataset. Just indicate names of the others in your report. Such a dataset will be expected to be of better quality.
    - You need to describe (1) your data source, (2) criteria for data inclusion, if any, and (3) criteria and method for labeling.

Algorithms:

You need to use at least four types of classifiers in your experiments, ranging from the really basic ones like kNN and logistic regression to more complicated ones like SVM and boosted classifiers. Make your choices. You can use existing libraries and public available implementations such as scikit-learn; just indicate in your report the source of the implementation used. A rule is that at most one of the four classifiers is CNN based.

Analysis:

Evaluate the performance of your classifiers using tools and metrics such as accuracies and confusion matrices. For yes/no classification problems, measures such as precision/recall/F1/AUROC should be considered.

- ■ Be sure to use cross-validation to obtain the evaluation metrics.
- ■ Compare the results with different amount of training data.
- ■ Compare the results among different classifiers.
- ■ Compare the results using different settings and/or hyper-parameters for the same classifier.
- ■ (Optional) Compare the results with and without dimensionality reduction (such as PCA) of the features when using high-dimensional data.
- ■ (Optional) Experiment with the effect of data augmentation.
- ■ For public datasets, compare with results given in the respective websites or literature if they are available. Provide references if you do so.

Discussion:

- ■ Based on your experiments, are the results and observed behaviors what you expect?
- ■ Discuss factors that affect the performance, including dataset characteristics.
- ■ Describe experiments that you would do if there were more time available.
- ■ Indicate what you have learned from the experiments and remaining questions.

Your submission should be a report file in PDF format. The report (maximum 10 pages single-spaced) should contain the sections listed above. Submit your report through E3. Late submission is accepted for up to 3 days, with a 10% deduction per day.

Include your program code as an appendix (not counting toward the 10-page limit), starting from a separate page. You can use C/C++ or Python to write your program. In general, the TAs will not actually compile or run your programs. The code listing is used to understand your thoughts during your implementation and to find problems if your results look strange. Therefore, the code listing should be well-organized and contain comments that help the readers understand your code; this will also affect your grade.