CS 1390-01: Introduction to Machine Learning

Projects List (Given Oct. 29, 2020; Selection Due by Nov. 4, 2020)
Projects Reports Due: December 6, 2020 (11:59 pm) – No Extensions

---

You are required to do a project in this course. A list of candidate projects are given below. Projects are executed by teams of 4 or fewer people. So, form a team and choose a project that appeals to you (each project will be executed by one team only and projects are chosen on a first come first served basis) latest by Nov. 4, 2020 and enter your team and project choice on the Google sheet located at `https://docs.google.com/spreadsheets/d/1t1MGLEYg4OLBNYiFRJC6uFKk2NAJ3m599dNUBa4kelw/edit?usp=sharing`. Each project is identified with the serial number in the Projects List PDDF file. If a project has names against it, it is taken. To select your project, you need to find a project which does not have names against it and enter the names of the complete team that will work it.

You need to submit a final report (PDF) and an executable copy of the source code for the project into Google Classroom. In the event, there is a problem with Google Classroom, you must email your submission with a time stamp which is not later than the submission deadline indicated above. There is sufficient time for you to plan your work and submit without waiting for the last minute. There will be no extensions.

Your project report should contain the following at the minimum (numbers in parentheses indicate the grading weightage),

1. A description of the problem and its significance (10%)
2. An overview of approaches that exist (20%)
3. The approach that you selected and why (10%)
4. The implementation details (25%)
5. Results obtained and comparison with state-of-the-art results (25%)
6. A discussion of the results (were they anticipated, if not why did it not) and Conclusion (10%)

Do not lift implementations from the web or copy past projects. That would amount to plagiarism. You will also need to demonstrate a working system to the TAs for full credit.

1. This project deals with trying to find changes in the facial expression of a person between pairs of images (ideally this is done with continually streaming video and changes between frames are detected; however given the limited computational resources you may assume that you will do it over images acquired at points in time). Any substantial shift must be detected irrespective of what the expression was before and what it is after the shift. You may assume that the photographs are facial and the head movement is small (not absent).

2. In this project, you are trying to predict future sales. Often, this is required to order inventory at the right time and often to decide what to discount to generate demand. Indeed there are many other use cases as well. Walmart `https://www.kaggle.com/fernandol/cracking-the-walmart-sales-forecasting-challenge` provides several data for 98 products across 45 outlets and you can access information on weekly sales by locations and departments. The goal of this project is to predict the sale of the 3 of the most selling and 3 of the least selling products.

3. This project deals with constructing a neural network that can automatically determine the age and gender of a human in an input image. There are a lot of training data sets available (see for example, `http://www.openu.ac.il/home/hassner/Adience/data.html`). Create provisions in your implementation to capture an image (perhaps using the camera in your laptop) and output the age and gender.

4. This project deals with recognizing facial expressions i.e. classify each facial expression into one of 7 emotion categories viz. angry, neutral, sad, happy, surprise, fear, disgust. There are lots of training data sets available (see for example, `https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data`). Create provisions in your implementation to capture an image (perhaps using the camera in your laptop) and output the emotion category.

5. This project deals with colorizing back-and-white images such that he colorized images appear natural. There is a lot of training data sets available (see for example, `http://vis-www.cs.umass.edu/lfw/`). Create provisions in your implementation to capture an image (perhaps using the camera in your laptop) in black-and-white and colorize it.

6. This project deals with building a crucial component of driverless cars viz. automatic traffic sign recognition. You can use the Belgium TS dataset for your purpose (`http://btsd.ethz.ch/shareddata/`). Create provisions to capture an image (perhaps using the camera in your laptop) in black-and-white and colorize it.

7. This project deals with verifying human signatures i.e. learn a set of (name, signature) pairs and then given a name and a signature, you should be able to say if the signature is authentic. You can use the dataset from `http://www.iapr-tc11.org/mediawiki/index.phptitle=ICDAR_2011_Signature_Verification_Competition_(SigComp2011)`. Create provisions to store an individual's (name, signature) pairs and then capture an image of a signature (perhaps using the camera in your laptop) and score its authenticity.

8. This project deals with recognizing a person from facial images. There are many datasets available for you to use and construct your network. See for example, `http://www.face-rec.org/databases/`. Create provisions to capture multiple images of a person (perhaps using the camera in your laptop) and then be able to recognize that person.

9. This project deals with recognizing handwritten ZIP (PIN) codes. There are many datasets available e.g. the MNIST dataset (`http://yann.lecun.com/exdb/mnist/`). However, there is a twist. You need to randomly change 20% of the squares (from black to white or vice versa) in each of the training data set patterns. The test data set patterns should remain unchanged. Create provisions to capture an images of a zip code written by hand (perhaps using the camera in your laptop) and then recognize the zip code.

10. This project deals with building a crucial component of driverless cars viz. vehicle detection on the road. You can use the dataset available at `http://cvrr.ucsd.edu/LISA/vehicledetection.html`.

11. This project deals with detecting hand gestures. You can use the dataset available at `http://www-vpu.eps.uam.es/DS/HGds/` or the one at `http://ee.sut.ac.ir/showcvmain.aspx?id=5`. Create provisions to capture a gesture (perhaps using the camera in your laptop) and then detecting the gesture.

12. Build a network to recognize objects. You will use the CIFAR-10 dataset which has 10 classes and is available at `http://www.cs.toronto.edu/~kriz/cifar.html`. Your submission must innovate on the state-of-the-art and must be able to beat it for full credit. Create provisions to accept new images and produce their classification.

13. This is also an object recognition project. Build a network to recognize fashion articles. You will use the Fashion-MNIST dataset which has 10 classes and is available at `https://github.com/zalandoresearch/fashion-mnist`. Your submission must innovate on the state-of-the-art and must be able to beat it for full credit. Create provisions to accept new images and produce their classification.

14. This project deals with detection of fraud in the financial domain. Though there are multiple approaches that are possible, the idea of this project is to use clustering to detect financial irregularity. You can use any of the multiple datasets that are available including the one at `https://www.kaggle.com/ntnu-testimon/paysim1`. Make provisions to accept a transaction to test your implementation.

15. This project deals with trying to find companies whose stock price behavior is most similar to a query company. To make things simpler, assume that the universe of companies are the top 100 companies (based on market capitalization) listed in the Bombay Stock Exchange and assume that everything starts from the year 2010 and any history prior to that can be discarded. So, if the query company is INFY, then you should retrieve one or more companies whose stock price behaves most similarly to INFY. How many to retrieve is a design parameter as is a definition of "similarity". Your interface should have a text box in which the name or the ticker symbol of the company can be typed and your program responds with the most similar companies as determined from the stock price behavior.

16. In this project you will try to cluster emails. The clustering can be based on any part and any representation of each email but the resultant clustering must make some sense. The dataset is based on the actual emails of the senior executives of Enron made public by FERC as part of its investigation and is available from `https://www.cs.cmu.edu/~./enron/`.