**CMPE 256 – Group Project**

**Project Timeline**

* 3/18: Project proposal DUE
  + Teams formed
  + Dataset selected
  + Submit idea
* 4/29: Project report & presentation slides due on Canvas
* 5/1 - 5/10 (tentative): Project presentations

\* This is a group project (groups of 3 students). However, each student will be graded individually (please refer to the green sheet for details).

\* A list of online resources is posted on Canvas.

\* All project documentation (project proposal, project report, datasets) should be submitted to the appropriate folder in Canvas (one per team).

**Project proposal**

Should be 1-2 pages long and include:

* Student names
* Team name (optional)
* Project title
* Project description
  + Dataset(s) to be used (name, link, size)
  + Proposed methodology/techniques (see details below)

It is important to secure an adequately large dataset before committing to an idea.

**General guidelines**

The project should be a machine learning/data mining application. The goal is to use appropriate algorithms for a specific problem, on a relatively big dataset, and to evaluate your options (same as you’d do in real life!). You should integrate/combine one or more algorithms/methodologies. In this project the team is responsible for picking a topic to work with.

You are provided a list of recommended datasets. Only up to 3-4 teams will be allowed to work with the same dataset (on a first come-first serve basis).

* Project deliverables:
  + source code, and/or instructions on how to install/use required software
  + experimental results (evaluation of algorithms, parameters etc.)
  + dataset, after pre-processing (if applicable)
  + front end/UI (if applicable, nothing fancy is required)
  + final report
  + presentation slides
* You may use publicly available (open source) code in your application and existing tools/platforms but this should be explicitly stated in your report. You can choose any programming language to implement your project. You are encouraged to use Spark’s MLib (Scala/Java/Python), Mahout on AWS (Java), or Python (and its machine learning/text/graph mining libraries), the various deep learning libraries and frameworks (e.g. Deepmask/Sharpmask, TensorFlow etc.) or implement everything from scratch. *WEKA GUI and other drag-and-drop software (e.g. Microsoft Azure, RapidMiner etc.) are* ***not*** *allowed for this assignment.*
* You are encouraged to use data exploration methods (e.g. visualization using WEKA, Tableau, Python etc.) in addition to your ML algorithms.

If you intend to use parts of a previous or existing project from other class (or your MS project), you need to talk to me in order to get approval (not guaranteed).

**Project Steps**

**Step 1: Pick a dataset and define your problem(s)**

You should use a dataset from the provided repositories. The dataset should be rich enough to help you perform various types of analyses and address interesting questions/problems (e.g. verify hypotheses, use machine learning algorithms to build predictive models or group together items, perform text mining/sentiment analysis, etc.).

**Step 2: Choose your algorithms**

You should pick appropriate algorithms and methodologies, depending on the problems you’ve set to solve. These can be among the ones we’ve already covered in class, or other machine learning/data mining techniques (e.g. sentiment analysis, social network analysis, graph mining, deep learning, etc.). Also, they don’t have to be all of the same type (e.g. 3 different classification algorithms) – instead you may use a combination of techniques to gain different insights into your dataset and problem.

There are no constraints as to what algorithms to use, but there are constraints on how many you should include in your analysis:

* Use at least 3 different algorithms (“hybrid” algorithms count too).
* Use at least 2 “tweaks” to each base algorithm (e.g. different parameter values, different pre-processing of data, etc.)
* Simple aggregations/statistical analysis will NOT count towards one of your 3 algorithms, but you may use them as needed in data exploration and preparation. Moreover, anything that can be answered by querying a database will NOT count towards your 3 algorithms.

A UI is not required and not a priority for this project. However, you may provide some demo of I/O or visualization of the results of your project, as applicable (this varies a lot depending on the project so please consult the instructor if in doubt).

**Step 3 – Compare and evaluate**

Use appropriate metrics to compare (where applicable) your approach and pick the “winning” solution.

Present your findings using graphs and verbal descriptions.

**Sample projects**

A few sample projects are (these are provided mainly to give you an idea of the scope of the project):

* Design a program that will classify review comments (on products, blogs, etc.) as spam/fake using sentiment analysis
* Design a recommendation engine that will generate recommendations taking into account other factors (e.g. social network/connections)
* Design a predictive model for urban development (e.g. identify who is building what in which parts of town – predict where the next housing boom will happen, etc.)
* Design a program that will use/analyze data from social networks. Note that there are so many possible ways of doing this – e.g. analyze tweets in different US states and find the “happiest” folks, then show results on a map.
* Develop your own system that will mine the social graph of a network and generate interesting patterns (e.g. important users).
* Use the data provided by a city (e.g. San Jose or San Francisco) and find interesting patterns & applications you can implement that would be useful in a community (e.g. environment-friendly).
* Design a recommendation system or perform some data analysis using fashion-inspired data (e.g. combine outfits). Deep learning could also be used in this context.
* Use data collected by smart devices (e.g. activity, geo-location, etc.) and design an application that helps the user in some way.
* ….

**Deliverables**

The projects will be evaluated based on significance, design, correctness, documentation, and appropriate evaluation/testing. You will need to submit your **project distribution files and documentation**, *both electronically and in hard copies* (CD/flash/USB drive & printout of the report). Your USB drives will not be returned.

A template for the project report will be provided to you. Roughly, your project report should contain the following components:

1. A description of your approach, including the data set used, problem solved, and algorithms/variations implemented and evaluated.
2. An evaluation of your system demonstrating its correctness and functionality. If you used any outside sources in your implementation, please clearly indicate which sources, and how and where they were used.

Your project distribution files should contain the following:

1. Complete source code (be sure that your source code is fully documented and easy to read).
2. Binary files (e.g., executables, DLLs, Class files) or other components necessary to run your program.
3. Readme file containing instructions on how to compile, install, and/or run your program.
4. Any test data used for evaluation of your system.

**Grading/Evaluation**

Each student will be evaluated individually. Your final grade will be calculated based on:

* The overall quality of the project report
* The overall quality of the project prototype (implementation)
* The overall quality of the experimental evaluation of your approach
* The individual presentation
* The individual participation in all project aspects (as shown by the project timeline and the peer evaluation)

**APPENDIX**

**Sample of previous CMPE 239 project topics**

Please note that this is for your reference, in order to understand the expected scope and extend of work. However, I won’t approve projects that cover the exact same topics (combination of idea & dataset) as listed below.

* Clustering analysis of Bay Area restaurants (dataset: Yelp)
* Restaurant success prediction (dataset: Yelp)
* Analysis & Prediction of Bike availability (dataset: Bike Share)
* Airline on-time performance analysis (SFO airport arrivals/departures dataset)
* SMS classification (dataset: SMS Spam from UCI repository)
* Identification of spam/fraudulent advertisements on craigslist (dataset: craigslist)
* Crime recognition and prediction system (dataset: SFPD crime incident database)
* Forecasting elections using sentiment analysis on tweets (dataset: Twitter).
* Polarity analysis of movie reviews (dataset: MovieLens)
* Finding smartphone trends using sentiment analysis on tweets (dataset: Twitter).
* Social information based Recommender system (project used social connections to make restaurant recommendations to users. Datasets: Facebook and Yelp)
* Personalized news feed using Facebook data (datasets: Facebook and Yahoo! News)