

Data Mining Assignment

NOTE:

- You only have to complete **ANY 2** of these suggested problems.
- I strongly suggest you use Python. Please do not use R.
- Feel free to use any Python library you wish.
- Please upload your assignments to Dropbox in Brightspace.
- My email address is sreejata@leadsift.com, in case you have any questions.
- Due date: First one by March 2nd, Second one by March 9th.

Assignment 1:

Download the sentiment labeled dataset of tweets:

<https://archive.ics.uci.edu/ml/machine-learning-databases/00331/>

1. Write a Naive Bayes Classifier for sentiment detection by training using above data (or any other labeled dataset you find)
2. Use at least two other classifier algorithms and report the difference in accuracy, tweak them so you get the best possible results.
3. Save the classifiers so you (and I) don't have to retrain every time. Please send me the code to read from the saved classifiers so I don't have to read them.

When submitting the code, the file that you want me to run should be called `firstname_lastname_any_other_description.py` and it should assume the training files, saved classifiers and any other required files are all in the same folder. It should read from the saved classifiers and not retrain it when I am running the code. Please include the results (accuracy) of the classifiers as a text file.

Notes: Take a look at Lecture 2, the classification code is provided. Feel free to use it as a starting point and add the other classifiers. [Here's another helpful tutorial on NLTK/Python.](#)

Assignment 2:

Use any data either from Twitter or one given by govt of Canada (or any other dataset you're interested in playing with) and figure out a data point that will surprise and impress you, your friends or me. Link to Canadian datasets: <http://open.canada.ca/data/en/dataset>

Upload the files/folder and clearly name the file that you want me to run as: your
firstname_lastname_any_other_description.py

Notes: This can be as simple as reading a file and finding the frequency or counts, averages etc. Or as complex as running clustering on interesting datasets to find patterns. The goal is to start understanding how to deal with large datasets and make sense of them in the real world, with undefined goals - what data scientists deal with.

Simple example: Which car gives the best mileage or has the least depreciation etc.

Interesting example: Read earning calls or nasdaq reports to predict which stock we should buy

Assignment 3:

Build a simplistic search engine. Please document the code well and write down what it does and doesn't do, what kind of ranking algorithm it uses, what data it searches on and what are the assumptions.

Upload the files/folder and clearly name the file that you want me to run as: your
firstname_lastname_any_other_description.py

Note: The goal here is to understand the workings of a search engine - please design the whole search engine and feel free to mock out the first or the last part so it. For example, just download some webpages (using curl or wget) and do the search on them instead of trying to crawl the whole web or crawl in real time You can also do a search on a single website (like SMU or Amazon). Do not bother with fancy display of search results. [Here is a good resource on building a Python search engine](#); but yours can be even simpler. Please include your design (a photo of your scratchpad is acceptable).

Assignment 4:

Use the Twitter API to get trending topics and then find some related news articles about those topics.

Upload the files/folder and clearly name the file that you want me to run as: your
firstname_lastname_any_other_description.py

*Notes: You don't *have to* use rss but it would be a good exercise to. It should give you different results if you run it at different times of day/different locations.*

Assignment 5:

This is a real “problem” that we at LeadSift are leading with. At LeadSift, we sell to businesses. (Ie, not consumers, so for example, Gillette or Pepsi sells to consumers, end users. Companies like KPMG/Deloitte or AWS etc, who sell to other businesses). Once we sell to a company, we want to sell to other companies exactly like that - look-alikes. There are some data available for download (Owler, AngelList, Bloomberg etc). Or, I can supply you with some sample data. Feel free to do your own research.

Upload the files/folder and clearly name the file that you want me to run as: your
firstname_lastname_any_other_description.py

Notes: This is a bit more involved and complex than the other ones and yet, this is the kind of problems you'll come across when you're in a work environment. You'll have a problem premise (find me more companies that I can sell to) and have to decide what datasets to use, how to find it and what algorithms to use to get an easy, fast and scalable solution. Happy to discuss if you wish to tackle this problem.

Feel free to email me any questions you may have.

All the best!