Project Proposal

Software Design Spring 2017

Mini Project 4

Minju Kang and Prava Dhulipalla

Possible project ideas (we need help to narrow down)

- Map of America or the world
    - Temperature data
    - Contour plot of elevation
    - Precipitation
- Political data visualization
    - Polarity of political speeches
    - Create a map of candidates' speaking about each other in the recent election
        - Positive and negative

Final project idea

> Create a map of the candidates' in election 2016 speaking about each other, with color-coded lines depicting positive talk, negative talk, and neutral talk. Possibly, there would an ability to switch between political spectrum to this positive/negative/neutral spectrum. The interactive portion would be being able to click on a name and see a randomly-generated sampling of quotes about the candidates, and perhaps a "positive," "neutral," and "negative" talk score.

The main idea of our project is interactive data collection. The topics that we will explore are: getting data from the internet, parsing large sections of data, creating graphic interface that's dynamic (responds to the reader, probably to the pointer on screen or any 'clicks' that may occur) and not just static ('just' an image), and visualizing data sets in a way that is understandable to the user. Our minimum viable product is a data visualization image that has, at the very least, 'something' the user can click on (for more information, anything interesting, etc). The stretch goal to have the entire data visualization interactive and dynamic - with pop-up boxes containing more information, being able to click and drag, and perhaps allowing the user to input something they want to see and only seeing that part (eg. a map and the user can put in their state).

Prava's learning goals are to learn about creating graphic interfaces in python and learning to work with a team member on larger-than-usual coding projects. Minju's learning goals are to extend knowledge about data mining, figuring out how to graphically represent data in an understandable manner, and how to build complex code.

We don’t know what libraries we are planning to use. To decide this, I think we’ll keep two things in mind: a) if the knowledge payoff is seemingly not enough for the time and effort we would have to put in, especially considering the time constraints for this project, we will probably use a library; b) if coding ourselves instead of using a library addresses some of our learning goals, we will probably not use a library.

By the mid-project check-in, we plan to have:

- We will have a skeleton created for the major classes and perhaps some of the functions that we need to create for the project. We will probably use the recommended UML diagram in order to do so - but probably try to be more specific.
- We will have divided up the tasks and set ‘due dates’ for each small part. (This is the implementation plan.)
- We will have researched relevant libraries and start to ‘map out’ where we will use libraries and where we won’t.
- We will start implementing functions and classes based on the schedule mentioned above.

The biggest risks that we see hindering our success is possible lack in python knowledge, time management (especially with other projects and homework assignments), possibly having our skeleton or schedule “be wrong,” and actually impede progress early on instead of helping us, personal and health issues that might lead to us not being able to work at 100% effort, and finally losing all motivation after QEA.