Logbook part 3

Presentations 1

1 Jordan and I presented first, on our project called APIEvolution, we received little feedback from the audience. However, during this presentation I talked about how our project which can predict stability through the history of an object could be applied to other areas (genetics, human tissue, communities, relationships)

2 Next was Devin Smith, with his presentation called: "Devins awesome spell checker" His project was cool, I liked how he applied the algorithms learned in class to something we use every day, like a spell checker. He used some of the same algorithms that Jordan and I used in our project, to detect string similarities such as levenshtein. He applied some other fixes to the algorithm to make the spellchecker work better.

The one thing I think that Devin missed out on which would have made his project so much stronger is a comparison with other spell checkers. In order to know how efficient/correct his implementation is, we need benchmarks, this is true for almost any academic work.

3 Lan presented *Model combinations in a recommender system*. I liked the motivating example at the beginning of Lan's presentation which was to find the correct movie to watch based on recommendations from a system. I think that Lan's talk was overall decent, but I don't think she took a *friendly* enough approach when explaining her data mining techniques like K-means for example. I understood, but only because I recently took the data mining class, I'm sure many people were not staying with her.

Presentations 2

1 Laura Mcleod's talk about *Reputation on StackOverflow* was a really interesting talk for me. Mostly because of the great relevance in the sense that I use that website every day, and am able to interpret her data on a personal level. She gave good motivation, and presentation of her results. The visualizations were key to her getting the audience engaged. I can see this work being adapted to any dataset which could include multiple users/entities which are interacting with objects that have *tags*.

2 Candy presented *Electric Load Prediciton* and I will admit that most of it was quite over my head, but I know that she * didn't say her name * needed to work on her public speaking a little more (speak loudly and clearly, as well as face the audience) On the other hand, she used very good graphics to display her results and capture the audience's attention.

3 Eirini Kalliamvakou presented *Scale Free Networks* which was a very interesting talk which seemed like she reached a really broad amount of network types and examples. She gave a really thorough analysis on the different graphs and methods for analyzing these graphs. So many references!

4 Nick had an interesting talk on *Procedurally Generated Melodies* which was by far my favourite so far. The way he used the sounds that the project actually produced to really connect with the audience was his greatest success in this talk. I think that the procedurally generated melodies are simply one area of using these models (modified Markov Models) to produce natural, probabilisitic results which are useful. This technique can be applied to any situation that is state based and has probabilities for the transitions (ex, C chord most likely follows a C chord).

Presentations 3

1 Zhuouli Xiao gave a talk called *Apply models to a small dataset: Nibbles & Bytes*. This talk was probably my least favourite because: * His descriptions of the algorithms were only in mathematical terms, use clear base case examples first. * Didn't have a good structure to his presentation (Having reoccurring main ideas throughout the slides is the best way to keep the audience on track) * He didn't present any related work or references for the project.

That being said, it is interesting that he used something local such as the data from Nibbles & Bytes.

2 Xuesong Yong presented on *Mobile Cloud Markov Chains* which was another topic that was slightly over my head,

Exam prep

EVERYTHING MUST BE DONE BY THE 8TH OF AUGUST

Take something that you don't know before and demonstrate that you learned a real skill

Questions

- Prove that you've learned something technical
-