**Criterion A: Planning**

The problem I am tackling for my IA is for the ComOps department at AEP. ComOps deals with the buying and selling of power. They needed a way to find news about the price of power and really all things that would interest that department on consumer sites. Google does not do this for them because Google a) does not do the best job of sorting by relevance in this instance, and b) does not write to a database. The client’s name is Joe Sheridan and he is in the ComOps department. A family member of mine has worked with him on a few occasions and that’s how I got in contact with him.

My Java program will search a list of sources for a given keyword and write the output to a database sorted by relevance. The sources will include most consumer news sites and will also include a Twitter API as well as the option to add more sources. All sources will be searched recursively but will not read duplicate links. Loops and recursion will be needed for searching. I will need to write to a database, likely Derby. GUI will be a very large part of this as I will not be the one using it. It must be intuitive and easy to follow. The initial output will be to an unsorted text file or csv file with 1 column for each source. Then, the data will be read into a class that will explore each link (when it is sure it is an article) and read the text and sort it. The sorting will be done in two ways: one, a ‘dumb’ classifier that reads the data and sorts it by the number of times the search keyword appears, and two, a ‘smart’ Python classifier. The neural network classifier will be in Python because of the wealth of libraries available to use. The dumb classifier will output to another text file with the links sorted in order. Then, the Python classifier will take that file as input and explore each link. It will compare the text to data in a test set, and sort it into 3 stages of relevance - Probably Relevant, Maybe Relevant, and Probably Not Relevant. This Neural Network will be trained using articles known to be relevant to the keyword that it had searched for. The classifier will write Probably Relevant and Maybe Relevant to the final database. The database will be able to be accessed by the user but in only read-only mode. The user will then use that data however they wish. There will however be an option to mark the webpage as relevant confirmed and then send that back to the training algorithm.

The goal of my program will be to get a list of webpages that could be potentially relevant to a certain keyword.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Task  # | Planned action | Planned outcome | Time estimated | Target completion date | Criterion |
| **1** | **Meet with Joe** | **Get project idea.**  **Stimulate thought** | **short** | **Completed may 2018** |  |
| **2** | **Meet with Joe** | **Get plan**  **Get enough to work on**  **for a while** | **1 hour** | **Completed August 2018** |  |
| **3** | **Work on project** | **get work done**  **No goal, just as much**  **as possible** | **15 hrs** | **8/23-8/30** |  |
| **4** | **Make test cases** |  |  |  |  |
| **5** | **Test test cases** |  |  |  |  |
| **6** | **Flowchart** |  |  |  |  |
| **7** | **UML diagram** |  |  |  |  |
| **8** | **Write psuedocode** |  |  |  |  |
| **9** | **Write real code** |  |  |  |  |
| **10** | **Make movie** |  |  |  |  |
| **11** | **Get feedback from client** |  |  |  |  |
| **12** | **Write-up** |  |  |  |  |

**Criterion C: Interview**

[Greetings]

Joe: Alright, so what is this school project.

Calvin: Well I’m taking IB computer science and we do a project with a client so I was wondering if you could maybe help me

Joe: Yes I probably could.

Calvin: Great! So do you have any problems you would need me to code a solution for?

Joe: Well we do a lot with news. We have commercial news sites that we use for getting things like the price of coal and stuff like that, but we don’t have anything to do that for consumer news sites. = **Problem**

Calvin: So what would that entail?

Joe: Well you would need to handle a lot of data and parse into links sorted by relevance. It would need to be stored in a database for the output. People on my team would use it so it would need to be easy to use. It would probably be a good idea to multithread because it’s a lot of data.

Calvin: Ok cool I’ll update you with my progress.

Joe: Sounds good.

[Calvin leaves]

Hi Calvin!

I hope you had a good summer! I spoke with my management earlier and we can only use data which is also available to the public, but that doesn’t mean we can’t still do something cool and useful to AEP. I had a couple of ideas, but I’m open to anything.

Com Ops uses a tremendous amount of information (for example news stories) from a large number of disparate sources. There is usually a lot of data presented that is of no use, but we still have to sift through it to find the few useful items. One idea I had was to write a program that could analyze various sites and produce a list of things we might be interested in. The most fun option might be to use the free (standard version) Twitter API (<https://developer.twitter.com/en/docs.html>), but you could always just pull in a couple of news sources instead. You could use a string compare or regex to find specific strings that would be of interest. Regular expressions are a little scary at first, but they are useful for a lot of different situations and worth learning. You mentioned that you were interested in machine learning, if you want to take that on you could use automatic classification tools (that will be a lot of work though). If you wanted to build a GUI, the user could input keywords to look for. You could write the results to an HTML file (maybe just links and a subject line) or even a database if you want to get ambitious. Just one idea, and if you like it we can flesh out the requirements.

Another option I thought of is to use the publicly available Data Miner 2 from the PJM website (<https://www.pjm.com/markets-and-operations/etools/data-miner-2.aspx>). PJM is a bulk electricity market that AEP sells our generation (energy created from our power plants) into and buys our load (the electricity our customers actually use). We can go over what the market is and isn’t if you want to go this route. Data Miner 2 has a web service API (<https://www.pjm.com/-/media/etools/data-miner-2/data-miner-2-api-guide.ashx?la=en>), though I’ve never used it so I’m not sure how complicated it is. Pretty much anything publicly available on Data Miner 2 would be useful. If you wanted you could store it into a database like derby (<https://db.apache.org/derby/>) or something.

Let me know what you think. I’m open to just about anything.

Thanks!

-Joe

Thanks for working to let me help you with some programs. I'm not really sure what the second option's goal would be, do you think you could elaborate a little on it? The first option looks like it would work well for me though, and I've done some regex stuff before in linux although I'll need to brush up a little on using that with Java. I also did some GUI work last year for a project so I could make a front end for it too. I'm interested in the ML automatic classification feature for that as well, and it would be really great to pipe the output to a database. What are the requirements for that project?

Also, when would you need me to come down to AEP next?

Thanks,

Calvin Kinateder

