LIN6209 Assignment 5
General Feedback to ALL Students

You have written your project proposal. Well done.

However. Do not let your proposal become a ball & chain.

Software design and development is HARD, and unforeseen difficulties lie in wait for you. That’s just the nature of the thing.

If your project becomes more difficult than you had imagined, then do not shy away from changing it.

Making a change to your project plan/proposal is best done earlier rather than later. Simplifying or changing your project proposal will take the pressure off and allow you to deliver results. Then, if you still have time, you can have a fresh go at the thing that was blocking you, this time maybe with new insights that allow you to move forward.

When designing software do remember the acronym KISS! = “Keep It Simple Silly/Stupid” and its many versions. Or, perhaps more appropriate to a software project is the variant by Bjarne Stroustrup: “Make Simple Tasks Simple!"

More than a few of you have done some online research into Python tools & libraries available for the analysis of text (which is good) but then succumbed to the temptation of designing the bulk of your project proposal around the use of those tools. The problem with this is that all those tools assume you have a good knowledge of the Python environment and excellent python fluency – which I more than suspect quite a few of you do not.

Starting your project by trying to get NLTK or Beautiful Soup or VADER or spaCy or sciKit-learn or SQL or whatever installed and working for the first time can be a challenge, and rapidly become a big time-sink. Time in which you could have built and delivered Python functions of your own design that would have delivered real value and demonstrable results.

I’m not telling you not to use those tools, on the contrary please do, but only after keeping the first iteration (or two or three) simple and uncluttered so that you can produce some results that demonstrate your fluency with basic Python. Then, safe in the knowledge that you already have good work in the bank, you can launch into adventurous territory feeling safe and confident that even if all this ambition goes pear-shaped, you and your project are still OK.

Do not shy away from doing simple things simply:

- Use cut & paste to get data from websites
- Clean your data manually in a text editor rather than programmatically
- Functions to read a piece of text and count letters, count scores, count words, and so on
- Calculate letter/word frequencies per 1000 letters/words
- Store those results in Python dictionaries
- Simple plots to display the patterns in those results.
- Comparative plots for two or more texts.

Several of you have talked about identifying nouns, verbs, adjectives, adverbs, and so on in your texts. Doing this programmatically is considerably harder than it looks. Do take a simple 'manual' approach in the first instance.

Do beware of techno-buzzword-soup. It might work in sales & marketing pamphlets; it does not work in your project.

DO think about test data that you can use to prove your functions are working correctly. For instance, to test your function count_letters(a_file) create a small file with known letter counts.

I'll speak more about 'doing' your project and writing your project report in our remaining lectures.

You have your project proposal. The time to start working on it is NOW.

Soonest started, soonest sorted!

Good luck 😊