Final Project Info
Ling 111

1 Topic

1.1 Overview

The final project will count for 30% of your grade, so you should begin thinking about it early on. Your project should apply some of the ideas and/or tools we used in class, either to answer a research question or to perform a task of practical utility (insofar as is possible within the scope of the course); it must also include some description of what it does, how it does it, and why what it does is interesting or useful. The project must show that you have learned enough about how to do computational linguistics to do something on your own that we didn't already do collectively in the class.

The project must be submitted on GauchoSpace by 8pm on Monday, March 18 (one week after the last class, just like a homework assignment).

You can draw on code that you used in the homework assignments, but your project must do something that goes beyond just re-doing what you've done in the homework assignments. That is, you *cannot* just take the code you used for the homework, run it again with different data, and call that a project. (However, combining techniques from mutliple assignments into some new creation might work.)

You can use one or more of the datasets we used in class, or the others provided on the website, or get your own data. Since this class is focused on dealing with fairly large datasets, your dataset should be sizable. A rule of thumb is that the dataset should be large enough that it would not be practical to even look at all of it, let alone manually analyze it. (An exception to this is if you work with data in a language other than English, both because that will be more challenging, and because it is somewhat harder to find suitable large datasets in that case.)

Within those broad guidelines, you are free to choose pretty much any topic and method you like! To ensure that your project is workable, **you must meet with the instructor or TA** to discuss your plans. You are advised to come to office hours as early in the quarter as possible. If you have an idea, we can talk about that; if you don't have an idea, we can discuss possibilities. Partway through the quarter, a more formal schedule of timeslots will be set up to ensure that everyone gets their project idea vetted.

1.2 No duplicate projects

No two people can do the exact same project. This means that you should get your project approved as early as possible before anyone else steals your idea. Of course, many projects may have similar topics, and you are encouraged to discuss your projects with other students, but the projects must be different enough that each person has to do their own work.

1.3 Format

You should create your project as a Jupyter notebook. It should contain your code, with comments where appropriate, as well as Markdown cells containing prose that walks the reader through the what, how, and why of that code. In other words, your project should not just be code, it should be like an article or essay that uses code to illustrate its points, similar to how a more traditional essay might use images or quotes from books.

1.4 Grading

Grading for the project will be broken down into four broad categories:

Approval (10 points) for meeting with the instructor or TA and geting your project idea approved.

Code (50 points) for having nicely commented code that does more or less what you wanted it to do.

Explanation (50 points) for having good textual description in Markdown cells explaining the project.

Content (40 points) for overall quality of your project topic and your treatment of it.

The "Content" category is meant to assess the overall thoroughness and coherence of your work. You will get a good score on Content if you explore your topic as fully as possible within the scope of the course, and if the integration between your code and textual explanation is good, so that the notebook is streamlined and can be read with a consistent "flow". Basically what this means is you will lose points on Content for things like: doing very little in your project beyond what we did in class; going off on a lot of tangents; having some bits of code and some explanation of what they do, but little explanation of why you've put together the pieces in this particular way.

1.5 Sample Topics

Some ideas for projects include but are not limited to:

- Write a chatbot that uses a somewhat more sophisticated Markov chain model.
- Write a Markov model that makes use of information besides n-gram frequencies (such as part-of-speech information or word vectors).
- Write a summarizer that makes use of spaCy word vectors. See how it compares to the frequency-based one. Talk about which one seems to work better and why.
- Write a summarizer that doesn't just use the word frequencies of the text being summarized, but compares them to baseline frequencies from some reference corpus.
- Compare performance of different sciki-learn models on a sentiment analysis task.
- Experiment with different approaches to preprocessing data before passing it to a scikit-learn sentiment analyzer.
- Try doing a sentiment-analysis-like model that predicts something other than good-bad judgement (e.g., one that tries to identify whether a Yelp business is a restaurant, or whether a movie review is about an action movie).
- Write a document-similarity tool that recommends authors with a similar writing style to a provided text.
- Write a plagiarism detector that compares sentences or paragraphs to Wikipedia articles.
- Write a Markov-based chatbot that uses speech recognition so it can respond to spoken words.

Remember that each person must do a different topic. But many of the ideas above are broad enough to support multiple projects.

1.6 Don't panic

It is *completely okay* if your project winds up not working as you intended. In that case, your commentary should explain what you were expecting, and what happened instead. Another possible idea for a project is to investigate two ways of doing something, and compare the results. Even if neither is all that great, you can still compare their strengths and weaknesses.

It's okay if your project has a bit of the feel of an explorer's journal rather than a completed research project. It *is* a notebook, after all!

Try, if possible, to have fun!