Data

MASC

We plan to start with English-only data for our first models (including common punctuation and whitespace characters). We will use large text datasets, starting with the MASC, a subset of the American National Corpus. It's a ~500k word dataset of public domain English text and licensed under the Creative Commons Attribution 3.0 United States License, making it acceptable for use in our project.

For each document in this corpus, some preprocessing such as removing excess whitespace will be necessary. Some files like court transcripts contain metadata about the text like date and author. We will likely remove this metadata for a cleaner text file.

This dataset is useful because it contains both written and spoken English text and can be easily obtained by anyone through a download.

Project Gutenberg

Project Gutenberg is an online library of public domain (in the US) eBooks. Although we plan to start with English books, it also has books in a variety of languages which we may use in the future to expand the model's language capabilities. The downside of this source is that books must be individually and manually downloaded. Additionally, public domain books tend to use older styles of writing since it takes time for books to become public domain, so the content of these books may or may not accurately represent current English writing styles.

These files contain metadata and chapter information at the top and licensing information at the bottom, which we will not feed into our model (or else it may be biased towards talking about US copyright law).

Methods

We will use Python for this project as it is what the group is comfortable with and it has many ML/NLP-related packages for use.

For the first checkpoint, we will fit a lookup table of all character n-grams (we will experiment with the value of n) seen in the text to the three most-common next characters. To make predictions, we will take the most recent n-gram from the input string and return the top-three next characters using the mapping.

This method does not capture the previous content of the input utterance, meaning it does not necessarily make better predictions with more context. We plan to improve upon this in future iterations.

Future Directions

In later iterations, we plan to experiment with neural network architectures (in pytorch).

One approch is token-based (as opposed to character-based) feature representations. We will explore semantic feature representations in hopes of making more-informed predictions. A simpler option is the bag of n-grams (n is a hyperparameter) feature representation.