Beau Carlborg
Clara Richter
Sunny Chiu
Data Science and Machine Learning

Implementation Summary

The Project

For our project, we are planning to use a data set of books to create a trained algorithm which will be able to predict the year that a book was written in. We plan to use Project Gutenberg, Google books, and Ngrams to perform this data analysis.

Gathering the data - gutenberg_catalog.csv, goodreads_catalog.csv

For this implementation, our group made significant headway on the data collection and data gathering elements of our project. For our project, we have been going about finding a collection of books to download for our dataset as well as finding the original publication date that corresponds to each work. The most significant task we have had to overcome in this iteration was finding accurate and reliable sources for the publication dates of our data. In the initial phases of the project, Sunny wrote a web scraper that was able to utilize project Gutenberg to download a book and retrieve a small section of data from that book that could then be analyzed using NLTK. Clara and Sunny both worked closely together on this spike story and laid the groundwork for having a deeper understanding of the nature of the data we will be encountering for the rest of this project. While they worked on scraping sections of the book, Beau wrote a webhook to the google books API that could be used to search a book title from Project Gutenberg, and get the earliest publication date of books matching that title from the google books API.

Throughout this initial phase of work, the majority of our energy was spent trying to develop a deeper understanding of the nature of our problem, and the data available for it. All three of us spent a significant amount of our time exploring possible resources for data and the publication date of the books. Once our respective initial web scrapers were complete, we moved on to attempt to create a catalog of publication dates for the books on Project Gutenberg. This process involved creating a parser which could read through the record of books from Project Gutenberg, then check the titles of those books against the google books API. The entire team worked very closely together on this story. The catalog of titles for project Gutenberg was very poorly formatted and lead to a significant amount of trouble for us. The catalog of titles was many tens of thousands of lines in a plain text file that was constructed from some table like software. Clara, Beau, and Sunny all worked closely together to create a parser for this catalog. Eventually, we were able to combine the google books API with the Gutenberg title Catalogue. Upon completing this merge though, we realized that the google books API was not generating accurate publication dates for the more obscure titles in the catalog of Gutenberg books.

Upon this realization, the group split up for a short period, each performing a spike story on a few other methods that we could use for obtaining the publication dates. Upon reconvening, the group decided to take an alternate approach forward. We decided to create yet another web scraper which could use information from goodreads.com to obtain the publication date. The Goodreads website provides a catalog of many of their favorite books in Project Gutenberg, and most importantly, provides the original publication date for each work. Upon realizing this, the whole team made an aggressive move forward to try and create a Goodreads web scraper that we could use to obtain a list of books and publications dates from. Sunny worked on the scraper which reads particular book pages on Goodreads. Beau worked on creating a headless python web-scraper which could be used get past the login page required by Goodreads. Clara wrote a new and improved parser for the Gutenberg catalog which was able to gather more data. When each of us reconvened, we were able to combine our work to create a program which was able to scrape Goodreads, combine the data of the publication date from the Goodreads, and retrieve the books from Gutenberg.

Visualizing the Data — project_notebook.ipynb

After a very extensive marathon of web-scraping and data collection, we finally were able to begin using the small set of data we gathered to visualize and analyze our data. As a group each member is not extremely familiar with the visualization tools we have available, so we all worked very closely together to make the visualizations in our project notebook.

The first visualization we created shows a line graph which displays the number of books we were able to find from each era using our data set. We decided not to show the entirety of our data set because the books we gathered came from a vast number of years, some of them as early as Plato and Marcus Aurelius. So instead, we decided only to show the number of books we have from the 19th century and onward.

The second plot we chose to make uses a single test book to lay the foundation for simple analysis of a single book. We used the book Pride and Prejudice as a test. We used this book to perform simple analysis on the factors like common words which were used in the book as well as other simple factors like average sentence length. We chose to use the results from this exploration to create a bar chart of the most commonly used words in the book. We decided to only visualize the 50 most frequent words because we felt that using more would only be overkill.

Finally, for the last of our visualizations, we chose to use the two variable factors in our project to visualize the change in the frequency that a word is used over time. To do this, we used 10 sample books from various decades and plotted the frequency of their use of the word “the.” This plot revealed small changes in frequency over time. But for the most part, suggests a relatively consistent word usage over time.

Running the Code — https://github.com/clararichter/dating-documents

Many of the scripts we wrote for this iteration were one time uses in order to scrape a website. In order to run our code, you only need to navigate to the directory you cloned the repository to and run “jupyter notebook ./project_notebook.ipynb”.