# Deliverable 1: Public Domain Book Dating

Garrett Kinman

## 1 Dataset

Language changes with time. But just how does it change with time? That is what this project seeks to answer. Or, more concretely, it seeks to predict the decade of a book's publication from the text itself. To achieve this, I will scrape Project Gutenberg's website for the plain text files and bibliographic information of public domain written works. I chose this dataset for three reasons: 1) public domain works are free and easily accessible, 2) Project Gutenberg is one of the largest hosts of public domain works available, and 3) Project Gutenberg offers the ebooks in plain text, which will make them much simpler to process.

## 2 Methodology

### 2.1 Data Preprocessing

The two biggest preprocessing tasks will be scraping the data from Project Gutenberg's website and parsing the text to cut out extraneous material. To do the former, I will need to both search through and access the plain text files of many different books—making sure to check titles so as to avoid duplicate works—and read the bibliographic information from the site itself. To do the latter, I will need to look for keywords and phrases that signify details of copyright and such. This is made simpler by the fact that these extraneous materials are at the beginnings and ends of each document.

### 2.2 Machine Learning Model

With this project, the goal is for the model to predict the decade of original publication of a public domain book from the text of the work itself. This means the model will have to handle a large number of features (books can have on the order of tens of thousands to over a hundred thousand words), and it will have to be a multiclass classifier (a limited number of discrete-valued decades to choose from). To this end, a convolutional neural network will be employed. Convolutional neural networks are often used in natural language processing, and are strong in multiclass classification and at handling large numbers of features. The downside is they can have poorer performance to, say, a naïve Bayes model, but I have access to a capable GPU that will hopefully be able to perform given a more intensive algorithm.

### 2.3 Final Conceptualization

In order to demo the final project, I will integrate it into a webapp. As I have experience with Angular—a framework that uses HTML, TypeScript, and SCSS— development, I will use that to create the front-end. The web app will allow the user to input a plain text file of an old book, and the program will return its predicted decade of publication.