1. Zipf's law and validation

Eetu Kiviniemi, Valtteri Impola, Sini-Sofia Korkeakangas

Project on GitHub:
`https://github.com/simppasofia/zipf-law-and-validation`

***Keywords*— Zipf's law, Python, NLTK, Linear regression, Power law with exponential cutoff.**

I. Introduction

According to Zipf's law, when ranked by frequency and arranged in descending order, the frequency *f(r)* of a word in any corpus should be approximately proportional to the inverse of its rank *r*.

$$f(r) \propto r^{-\alpha}$$

Zipf's law has been observed to apply to all languages.[1] A similar relationship can be noticed for many other naturally occurring phenomena as well, such as city populations[7], income rankings or people watching television[8].

The goal of this project is to study whether Zipf's law is applicable to varying corpora. Different statistical models will be fitted on the data, and the goodness-of-fit of the models will be mostly estimated based on graphical examinations. The Kolmigorov-Smirnov test is used to quantitatively test the goodness-of-fit of the power law model. Linear regression models will also be fitted for different subgroups of the corpora, such as only the words matching the topic of the books.
To ensure variability of topics, three books were chosen from separate subject categories; mathematics, history and technology. The chosen books are displayed in Table 1.
Flatland is a satirical novella, which uses a two-dimensional world of Flatland inhabited by geometric figures to comment on Victorian culture and society. The book was written by Edwin Abbot Abbot, an English schoolmaster and a priest, who is best known as the author of Flatland.

TABLE 1 - Basic information about the books

Author	Book name	Year	Book length
Edwin Abbott Abbott	Flatland: A Romance of Many Dimensions*	1884	25417
Ulysses S. Grant	Personal Memoirs of U. S. Grant, Complete*	1929	218127
J. Saxon Mills	The Panama Canal: A history and description of the enterprise*	1885	47200

** From here on we mostly use the following short names for the books to avoid repetition; Flatland, Ulysses and Panama.*

Table 1 - The authors and complete book names as well as the years of publication and the number of words in each book.

Ulysses is an autobiography by the 18th United States president and a general of the United States army during the civil war Ulysses S. Grant.The book is based on the memoirs of Grant himself, and was compiled into a complete book by David Widger in 1929.

Panama is a nonfiction book about the development and history behind Panama canal written by J. Saxon Mills. The book describes events relating to the canal starting from the 1500's, and ends in 1915, when the canal is opened.

The choice in books was also limited by availability on Gutenberg[6], an online library of free eBooks, which was used as a source for the books in this project. The books are all of similar age due to this limitation, but that does not pose any issues during the analysis.

II. Methodology

Before starting, the books were downloaded from the Gutenberg library as text files, and the preambles and postambles added by Gutenberg were removed in order for the calculations to be as accurate as possible.

A. Tokenization and data preparation

The books were tokenized using two alternative approaches depending on the task at hand. One was based on the methods available in the *NLTK* package, and the other was based on the findall() function available in the *re* package. All of the calculations were done using Python.

Once the books were tokenized, the ranks and frequencies of the tokens were calculated following an example given by Finn Årup Nielsen in his blog[2]. Token ranks and rank frequencies were calculated using the Counter() function from the package *collections*. The most common token would have the rank 1, the second would have a rank of 2 and so on. In case of multiple words with the same frequency, they would each be given a distinct rank in reverse alphabetical order. The ranks and frequencies that were achieved this way were used for many of the tasks done in this project.

B. Most Common Words

For each book twenty of the most frequent words, excluding stopwords, were identified using functions from the NLTK package and plotted on a bar chart in Figure y.

C. Linear Regression and Confidence Intervals

If the tokens follow Zipf's law, the frequency f of a token of rank r is given by

$$f(r) = \frac{C}{r^{-\alpha}},$$

where C and α are constants. For the logarithms of f(r) and r, this is equivalent to

$$log(f(r)) = C' + \alpha log(r),$$

where C' and α are constants. Therefore, if Zipf's law holds, a linear relationship between log(f(r)) and log(r) should be found. To investigate this, a least squares linear regression was calculated for the logarithms of the frequencies and ranks of frequencies of the tokens for each book with and without stopwords using the linregress() function from scipy.stats package as follows:

```
b1, b0, r_value, p_value, std_err =
stats.linregress(log(ranks),log(frequencies)).
```

The function returns the slope, intercept, and standard error of the linear regression. The correlation coefficient and p-value are not used in this analysis. Fitted frequencies $\widehat{f_i}$ for the range of ranks for each book i were calculated with

$$\widehat{f_i}(r) = b_0 + b_1 \cdot k.$$

Confidence intervals at confidence levels of 100(1-γ)% were calculated for each of the fitted values to investigate the goodness of fit of the model as follows

$$[\widehat{f_i} \pm t_{1-\gamma/2}(n-2) \cdot SE(\widehat{f_i})],$$

where $t_{1-\gamma/2}(n-2)$ is a fractile from Student's t-distribution and $SE(\widehat{f_i})$ is the standard error of $\widehat{f_i}$, which is calculated with

$$SE(\widehat{f_i}) = S\sqrt{\frac{1}{n} + \frac{r_i - \bar{r}^2}{SSX}},$$

where

$$S = \sum_{i=i}^{n} \frac{(f_i - \bar{f})^2}{n-2} \text{ and } SSX = \sum_{i=1}^{n} (r_i - \bar{r})^2$$

are the standard deviation of frequencies and square sum of the distance between ranks and their mean.

The linear regression fitted values and the corresponding confidence intervals at 95% confidence levels are shown with the points in a loglog-plot in Figure A1. The number of points that fell within the confidence intervals were also calculated, and are shown in Table 3.

D. Zipf distribution

Maximum likelihood estimate for discrete data was used to calculate an estimate for shape parameter α as follows:[4]

$$\hat{\alpha} \simeq 1 + n \left(\sum_{i=1}^{n} ln \frac{x_i}{x_{min} - \frac{1}{2}} \right)^{-1}$$

For x_{min} the minimum rank was used as this seemed to provide a reasonably good fit. It should be noted though, that this might cause some bias. A value for x_{min} is often chosen with visual methods.

After estimation Zipf curve was fitted using the Scipy premade zipf function.

E. Power-law with Exponential Cutoff

Power law with exponential cutoff (also known as truncated power law) is a power law function multiplied by an exponential term as seen below:

$$f(r) \propto r^{-\alpha} e^{-\beta r}.$$

Power law with exponential cutoff was fitted in order to explore a better fit for tokens with a low rank. Since there is no reliable closed-form estimator for α and β parameters for power law with exponential cutoff, the parameters were estimated with scipy optimizer package with good results.

Before passing the data to the `optimizer.curve_fit` function, the frequencies were normalized with min-max normalization in order for the optimizer to be able to find estimates for α and β.

F. Word Sense Variation

In order to get a number of senses per word, NLTK corpus reader WordNet was used. WordNet is a lexical database of English which consists of 117 000 synsets which are used to find conceptual relationships between words such as synonyms. WordNet works by grouping words together by meaning and therefore all words close to one another are semantically disambiguates [3]. After the number of senses per word was calculated. Results were plotted to chart using a number of senses and rank.

G. Word Categorisation

In order to test whether Zipf's law would hold for words in a certain category, appropriate categories based on the topics of the books were created in two different ways. The goal was to find a list of words from a category, and pick out matching tokens from the books. The matching words can then be plotted and the results graphically examined.

H. Categorisation Using Empath

Empath is a Python tool for analyzing text across lexical categories and generating new lexical categories to use for analysis. New categories are generated by giving Empath a few keywords, after which Empath retrieves a larger number of words similar in topic to the keywords. The keywords used for each book were *"colors", "shapes", "geometry", "dimension"* and *"line"* for Flatland, *"general", "military", "history", "army"* and *"soldier"* for Ulysses and *"history", "geography", "south america", "canal"* and *"jungle"* for Panama. Empath was used to generate a list of up to 400 words based on the keywords for each book. There are three possible sources for the creation of categories in Empath, in this case "fiction" was used for all books.

The books were tokenized, the stopwords were removed and the tokens were lemmatized using NLTK tools. The generated categories were then compared to the lemmatized tokens from the books, and matches were picked out. The points were plotted on loglog scale and on linear scale, where linear regression and 95% confidence intervals were also plotted on the logarithms of the points using linregress() function from the Seaborn package in Python.

I. Categorisation Using General Inquirer Categories

General Inquirer was found to provide relevant word categories for two of the books; *Personal Memoirs of U. S. Grant* and *The Panama Canal: A history and description of the enterprise.* For the

TABLE 2 - GENERAL INQUIRER CATEGORIES USED

Book name	GI category suggestions	Identifiable themes
Flatland: A Romance of Many Dimensions	1. Pleasur 2. Feel 3. MALE 4. Female	Humor, geometry, space, nature, relationships, gender
Personal Memoirs of U. S. Grant, Complete	**1. Milit*** 2. Legal 3. POLIT	History, military
The Panama Canal: A history and description of the enterprise	1. Work **2. Econ@***	History, construction, engineering

** Bolded categories were chosen for the categorisation task.*

Table 2 - The General Inquirer categories and identifiable themes for each book

third book *Flatland: A Romance of Many Dimensions* no appropriate category was found from these premade categories and therefore the previously introduced tool Empath was used to find categories for this book. General Inquirer categories that were considered for each book are listed in the Table 2.

After word categories were identified, both the tokens for each book and each GI word category were stemmed and matching words picked to represent a certain category in a book. The frequencies of the words were then plotted as advised in the instructions.

J. Graphical User Interface

Graphical user interface was generated using the tkinter package which is standard Python interface to the Tk GUI toolkit. This way of implementing user interface was selected due to it being built in feature of Python and therefore making simple UI development fast without need of heavy frameworks and fact that it is widely used when creating UI:s for Python applications . The selection of tkinter was also affected by the fact that there was no need for a separate back-end and API layer which would have been needed to be created if for example JavaScript would have been used to create a front-end with Python back-end.

Tkinter contains built- in widgets like buttons, frames and dropdown menus, which can be used to implement UI design faster because all UI components were already created in the library.

Application created during the project is divided into two sections, UI part which handles displaying data and interactions with user and business logic layer where all data processing is done for example filtering stopwords. This split was done to make code more readable and easier to manage.

III. RESULTS AND DISCUSSIONS

A. Most Frequent Words

The twenty most frequent words for each book with the stopwords excluded are presented as a bar chart in Figure 1. These word lists should be expected to match the topic of the book, and in reverse, at least a rough topic of the books can be deduced based on the word lists.

For Flatland, the twenty most frequent words were: *'one', 'would', 'see', 'line', 'flatland', 'two', 'three', 'must', 'could', 'even', 'circle', 'sides', 'square', 'every', 'may', 'say', 'sphere', 'space', 'straight'* and *'women*'. Many of the words are shapes and numbers, which is compatible with the topic of the book.

For Personal Memoirs of Ulysses S. Grant, the most frequent words were *'general', 'enemy', 'would', 'army', 'troops', 'could', 'one', 'time', 'men', 'river', 'two', 'command', 'sherman', 'left', 'corps', 'south', 'north', 'road', 'made'* and *'back'*. Most of the words are related to the military, which is to be expected based on the topic of the book.

For The Panama Canal: A history and description of the enterprise, *'canal', 'panama', 'states', 'united', 'new', 'would', 'isthmus', 'american', 'one', 'great', 'pacific',*

'may', 'sea', 'miles', 'west', 'feet', 'two', 'level', 'work' and *'america'*. The topic for this book was perhaps the most specific, and the list of words backs that up - the topic can be easily guessed based on even just a few first ones.

FIGURE 1 - BAR CHARTS FOR TWENTY OF THE MOST COMMON WORDS IN EACH BOOK

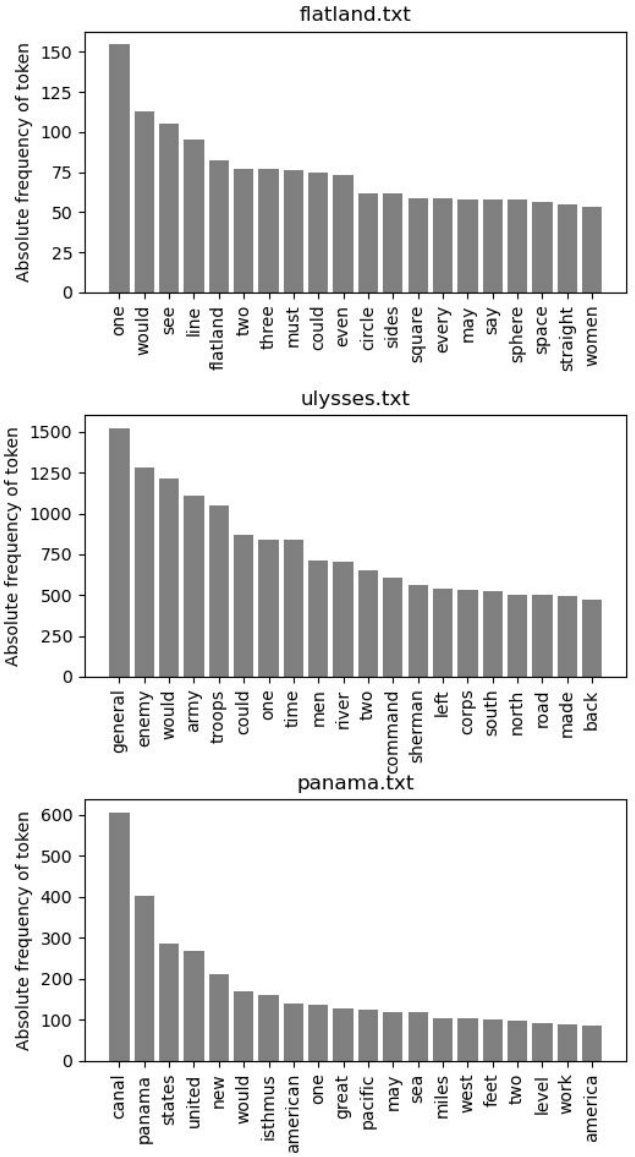

Figure 1 - Bar charts depicting the twenty most common words in each book, with the stopwords excluded. Y-axis gives the number of occurrences of each token, and x-axis depicts the tokens

B. Linear Regression and confidence intervals

When investigating the results of the linear regression seen in Figure A1 (appendix), we notice that the tokens do indeed follow a somewhat linear path on the loglog-scale plots, depicting the token frequencies as a function of ranks with an added linear regression line for the fitted values and confidence intervals for the fitted values.

TABLE 3 - NUMBER OF TOKENS FALLING INSIDE THE 95% CONFIDENCE INTERVAL

Book	Tokens	Points in 95% CI	% of points in 95% CI
Flatland	4515 (4418)	140 (109)	3.1% (2.5%)
Ulysses	9556 (9453)	189 (220)	2.0% (2.3%)
Panama	6388 (6293)	150 (123)	2.4% (2.0%)

Table 3 - The numbers and percentages of the tokens falling inside the 95% confidence intervals for the fitted regression lines for each book with stopwords included and excluded (in parentheses).

A significant decline on the most common tokens of each book when comparing the datapoints to the fitted linear line. The decline is, however, only significant for the first few hundred tokens out of thousands. Most of the points follow the regression line somewhat closely, because the large number of tokens with low frequencies pull down the slope of the regression line, forcing it away from the few high frequency tokens and closer to the many low frequency tokens.

Another slight deviation happens with the more infrequent tokens, where the datapoints tend to climb above the regression line. This can be best seen with the Ulysses, where a large portion of the mid-ranked points are above the regression line. The removal of stopwords doesn't improve the situation significantly either.

In fact, based on Table 3, it seems to make the fit worse for Flatland and Panama, and only slightly better for Ulysses. This is a rather surprising result, as one might have expected that there would have been significant improvements due to this, or in the very least for the effect to be in the same direction for all the books.

Ideally, the 95% confidence interval should contain 95% of the datapoints. However, in this case only a few percent of the points fall inside the intervals even at 95% with any of the books, as can be seen from Table 3. The main reason behind the low hit rate is that due to the large amount of data the interval ends up being quite narrow, and doesn't catch many of the points of a given frequency, even if the regression line passes through between them. Figure 3 depicts a zoomed in part of the Unfiltered Flatland graph. The confidence interval and the points around it can now be seen more clearly.

Flatland was the "most linear" of the books in the sense that it had the highest number of tokens fall inside the confidence interval, as can be seen in Table 3. Based on the plots in Figure 2, it does indeed look like Flatland follows the regression line more closely than the other two books.

FIGURE 2 - ZOOMED IN PART OF THE LINEAR REGRESSION GRAPH FOR FLATLAND

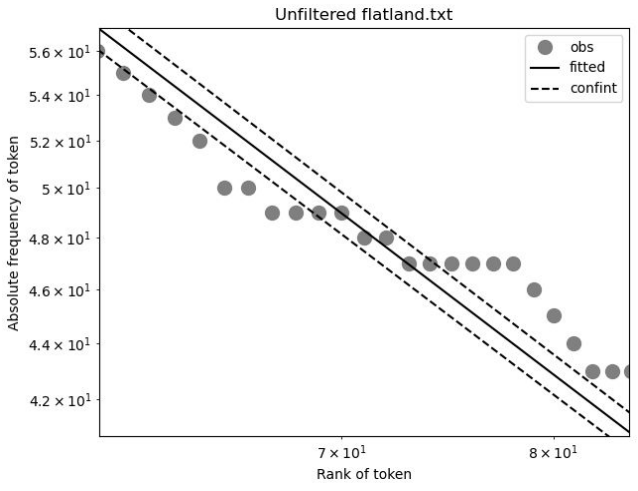

Figure 2 - Zoomed in part of the first plot in Figure 2, showing more clearly the confidence intervals

C. *Zipf distribution*

In general, the Zipf curve seems to provide a somewhat more realistic fit in comparison to linear regression. However, the fit seems to vary between ebooks. For Panama and Ulysses fitted values seem to not deviate from observed values as much as in Flatland's case.

Finding an optimal x_{min} value for the estimator of the shape parameter α could help in getting a better fit. This is discussed further in chapter IV.

FIGURE 3 - FITTED ZIPF COMPARED TO FITTED LINEAR REGRESSION

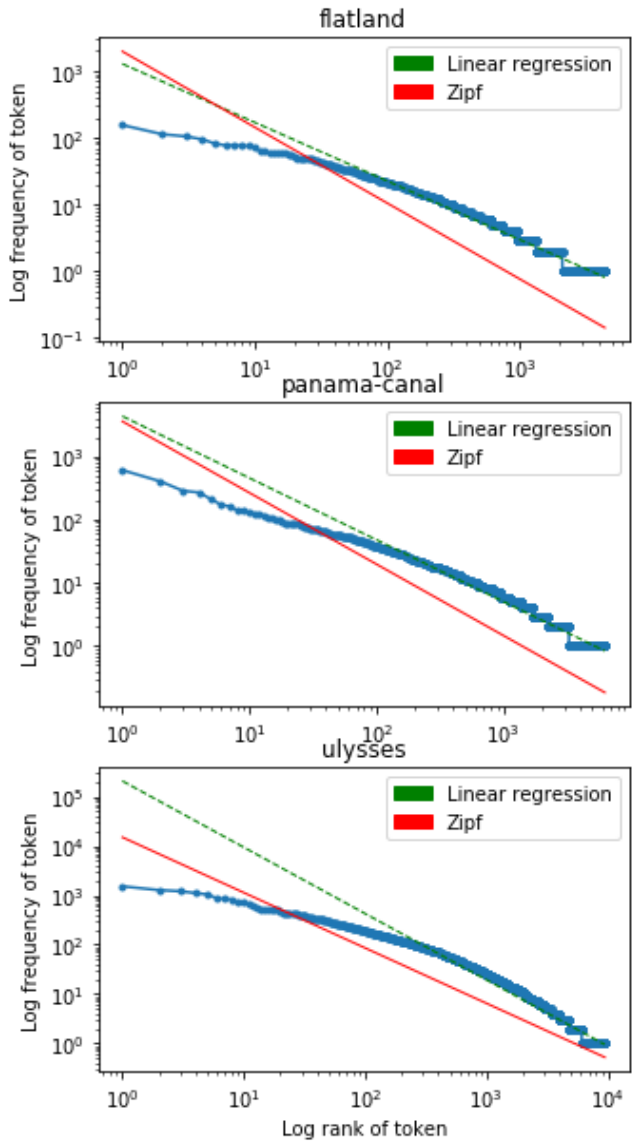

Figure 3 - Results for Zipf curve fitting

The shape parameter estimates achieved with maximum likelihood estimation are listed in Table 4. We can see that all of these are approximately 1.

TABLE 4 - ESTIMATED VALUES OF SHAPE PARAMETER

Book name	α
Flatland	1.135
Panama	1.129
Ulysses	1.123

D. Power-law with Exponential Cutoff

Results for maximum likelihood estimator for shape parameter α are as follows:

TABLE 5 - ESTIMATED VALUES OF PARAMETERS FOR EACH BOOK

Book	α	β
Flatland	0.386	0.00165
Panama	0.617	0.000517
Ulysses	0.402	0.00155

Visualization methods are the most popular form of goodness of fit validation for power laws and some methods for quantitative validation have been proposed such as the Kolmogorov-Smirnov test and likelihood ratio test.[4] To attempt to quantitatively assess the validity of the exponential cutoff we performed the Kolmigorov-Smirnov with Scipy's `stats.kstest` function for comparison of observed data and fitted data.[9] Achieved KS statistics are shown in Table 6.

TABLE 6 - KOLMIGOROV-SMIRNOV STATISTICS FOR THE BOOKS

Book	KS Statistic
Flatland	0.71
Panama	0.54
Ulysses	0.72

A higher value should represent a better fit (with values between [0, 1]). Judging by the test, the fit should be better for Flatland and Ulysses. However, this is quite a naive approach to quantifying the goodness of fit and does not give any information about how the power law with exponential cutoff compares to the other models.

FIGURE 4 - RESULTS OF FITTING AND COMPARISON TO ZIPF DISTRIBUTION AND LINEAR REGRESSION

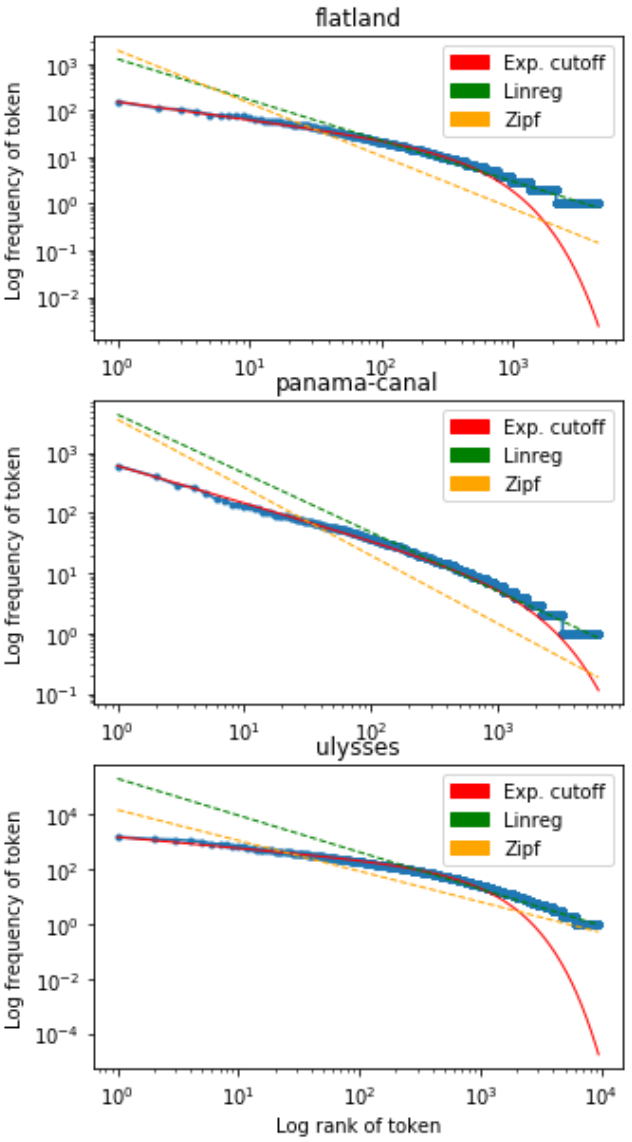

Figure 4 - Fit of power law with exponential cutoff represented by red curve.

The fitted curve seems to deviate quite a lot from the lower frequency words but of all the three models power law with exponential cutoff provides the best fit for a large range of frequencies.

The visual fit and statistic both seem to suggest that it is realistic to assume the data was generated according to power law with exponential cutoff.

E. Word Sense Variation

When investigating the results of word sense variation we can see that the number of senses seems to be following power law where most of the words have only few senses and only a small number of words have over 10 senses. Most of the words fall into that middleground. Linear fitting seems to be quite an accurate method of describing the data.

One observation from used method of WordNet was that it seems to be quite heavy method and when used on laptop seems to take a while to run and when used with tkinter application it sometimes lead into program freezing especially with books like Personal Memoirs of Ulysses S. Grant which contains almost double the amount of tokens compared to Flatland which runs quite fast. So an external server should be used to handle number senses calculation instead of using only the user's own computer.

F. Word categorisation with Empath

The numbers of total and unique matches between the categories and lemmatized books are listed in Table 6. A graphical representation of the results of this analysis can be seen on Figure A2 (appendix). As can be seen from Table 7, the total number of matches is much higher than the number of unique matches. The difference is most pronounced with Ulysses, which has almost hundred times more total matches than unique matches, four of the words appearing over a thousand times as can be seen from Figure A2.

The points for the matched token frequencies form curves shaped similar to the ones created with all the tokens. This is perhaps to be expected, as the matched tokens are just a subgroup of all the tokens.

Flatland was again the most linear of the three books with a majority of the datapoints falling inside the confidence limits. The curves for Ulysses and Panama were more curvy, and the lower-than-expected frequencies for the most common tokens were observed again. The confidence intervals for the matched token frequencies were wider than the ones calculated for the regression lines with all the tokens included due to a lower number of tokens being used. Thus a higher proportion of points is now contained inside the confidence intervals, but this is only due to the widened interval and does not necessarily mean that the tokens follow Zipf's law more closely than the books in their entirety for example.

TABLE 7 - NUMBER OF TOTAL AND UNIQUE MATCHES TO THE EMPATH CATEGORIES FOR EACH BOOK

Book	Total Matches	Unique Matches
Flatland	842	50
Ulysses	11732	129
Panama	1948	104

G. Word categorisation with General Inquirer categories

Total number of matching tokens by category for Panama was 460 and for Ulysses 134.

As can be seen from Figures 5 and 6, the fit isn't very realistic for either of the books' categories and we concluded that with smaller sample sizes the data does not follow a Zipf distribution.

It should be noticed that sample sizes used here are fairly small and the maximum likelihood estimator for shape parameter α is only accurate asymptotically which affects the fit.

FIGURE 5 - GENERAL INQUIRER CATEGORISATION AND FREQUENCIES

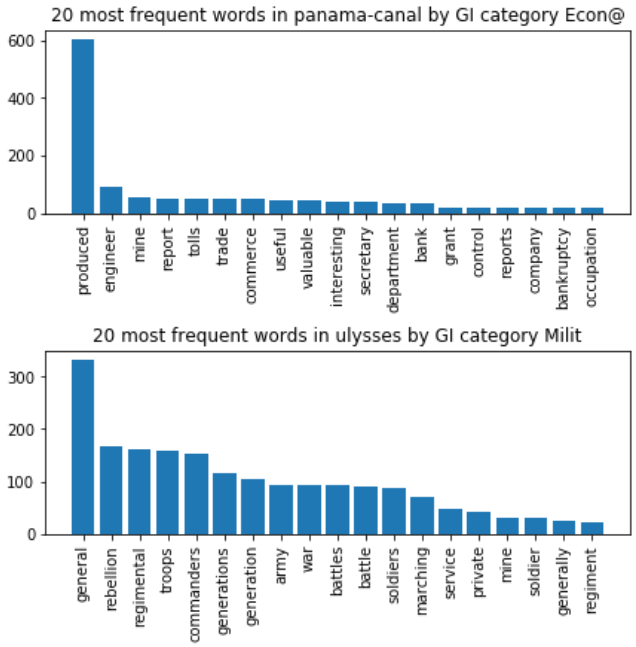

Figure 5 - A histogram for 20 most frequent words by General Inquirer category.

FIGURE 6 - ZIPF PLOT FOR GENERAL INQUIRER CATEGORIES

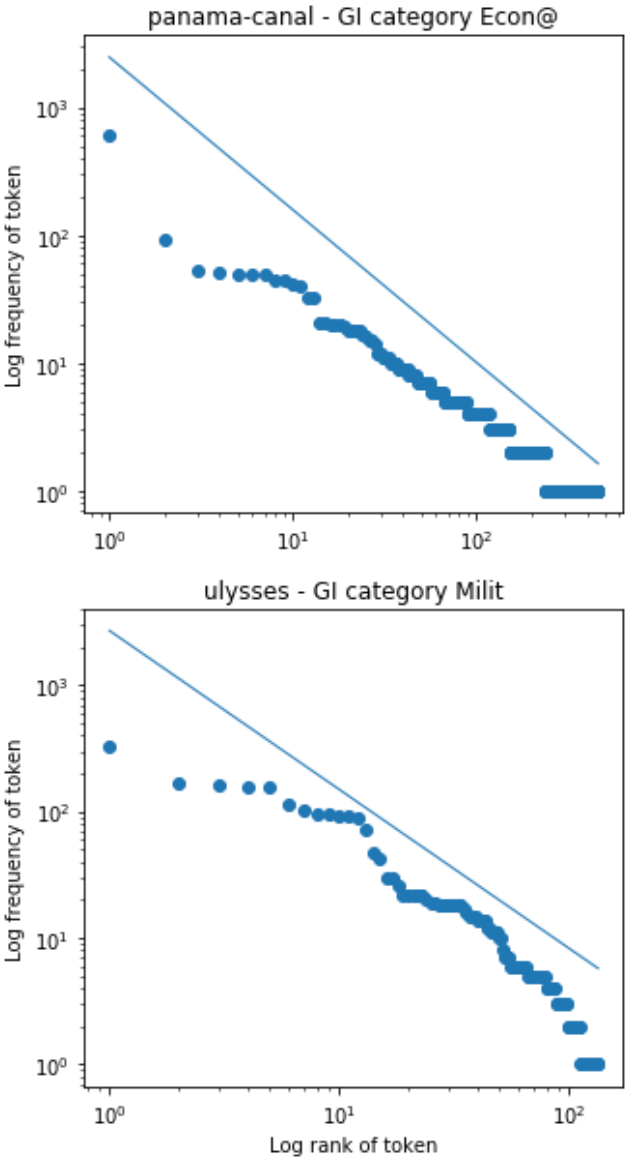

Figure 6 - Observations and fitted Zipf curve by General Inquirer category.

IV. Overall Discussions

A. Linear model for grouped ranks

One possible fix for the poor performance of the linear regression would be to group up the tokens with the same frequency into one datapoint. This would remove a lot of the probability weight from the lower rank end of the spectrum, thus allowing for a higher slope coefficient for the linear regression line. The following reduction in points would also increase the uncertainty of the analysis, widening the confidence intervals. This procedure may however make the interpretation of the graphs more difficult and remove some of the original information.

B. Fitting of Zipf distribution

In the future, instead of using the actual minimum value, the value for x_{min} when estimating shape parameter α for Zipf distribution could be chosen from the data by trying to find a limit beyond which the loglog plot is linear as advised in [4]. In this project such value that would have provided a better fit than the actual minimum was not found. This might reduce bias in the shape parameter estimate and provide an even better fit.

C. General Inquirer categorisation

Matched General Inquirer categories resulted in relatively small categories. When choosing categories we considered combining them to find more matches but were worried that this would result in categories that were less defined and would always end up in a very similar distribution as the whole data. Optimally the categories should describe the books well enough to produce a reasonable amount of matches. In the future another way of achieving this could be crawling the most frequent words of relevant web pages, e.g. Wikipedia articles of relevant topics.

V. Conclusions

While a perfect linear relationship was not observed for the frequencies and ranks of the corpora used in this study, it would be reasonable to say Zipf's law applies to these books. The law does not work as well with smaller corpora, such as the books while excluding stopwords or the topic category matched tokens.

Instead of a linear model, a more sophisticated model such as a power law with exponential cutoff model might be used to describe the data, if accurate fitted values are desired for the more common tokens.

VI. References

[1] Zipf's word frequency law in natural language: A critical review and future directions, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4176592/

[2] Finn Årup Nielsen's Blog: Zipf plot for word counts in Brown corpus, https://finnaarupnielsen.wordpress.com/2013/10/22/zipf-plot-for-word-counts-in-brown-corpus/

[3] What is WordNet?, https://wordnet.princeton.edu/

[4] Power-law distributions in empirical data, https://wiki.santafe.edu/images/6/66/CSN_07_PowerlawDistributionsInEmpiricalData_arxiv.pdf

[5] Descriptions of Inquirer Categories and Use of Inquirer Dictionaries, http://www.wjh.harvard.edu/~inquirer/homecat.htm

[6] Project Gutenberg, http://www.gutenberg.org/

[7] Auerbach F. (1913) Das Gesetz der Bevölkerungskonzentration. Petermann's Geographische Mitteilungen 59, 74–76

[8] M. Eriksson, S.M. Hasibur Rahman, F. Fraille, M. Sjöström, Efficient Interactive Multicast over DVB-T2 - Utilizing Dynamic SFNs and PARPS Archived 2014-05-02 at the Wayback Machine, 2013 IEEE International Conference on Computer and Information Technology (BMSB'13), London, UK, June 2013.

[9] Fitting power law distributions to data, University of California Berkeley, https://www.stat.berkeley.edu/~aldous/Research/Ugrad/Willy_Lai.pdf

VII Appendix

Figure A1 - Loglog-plots for tokens and linear regression lines with confidence intervals

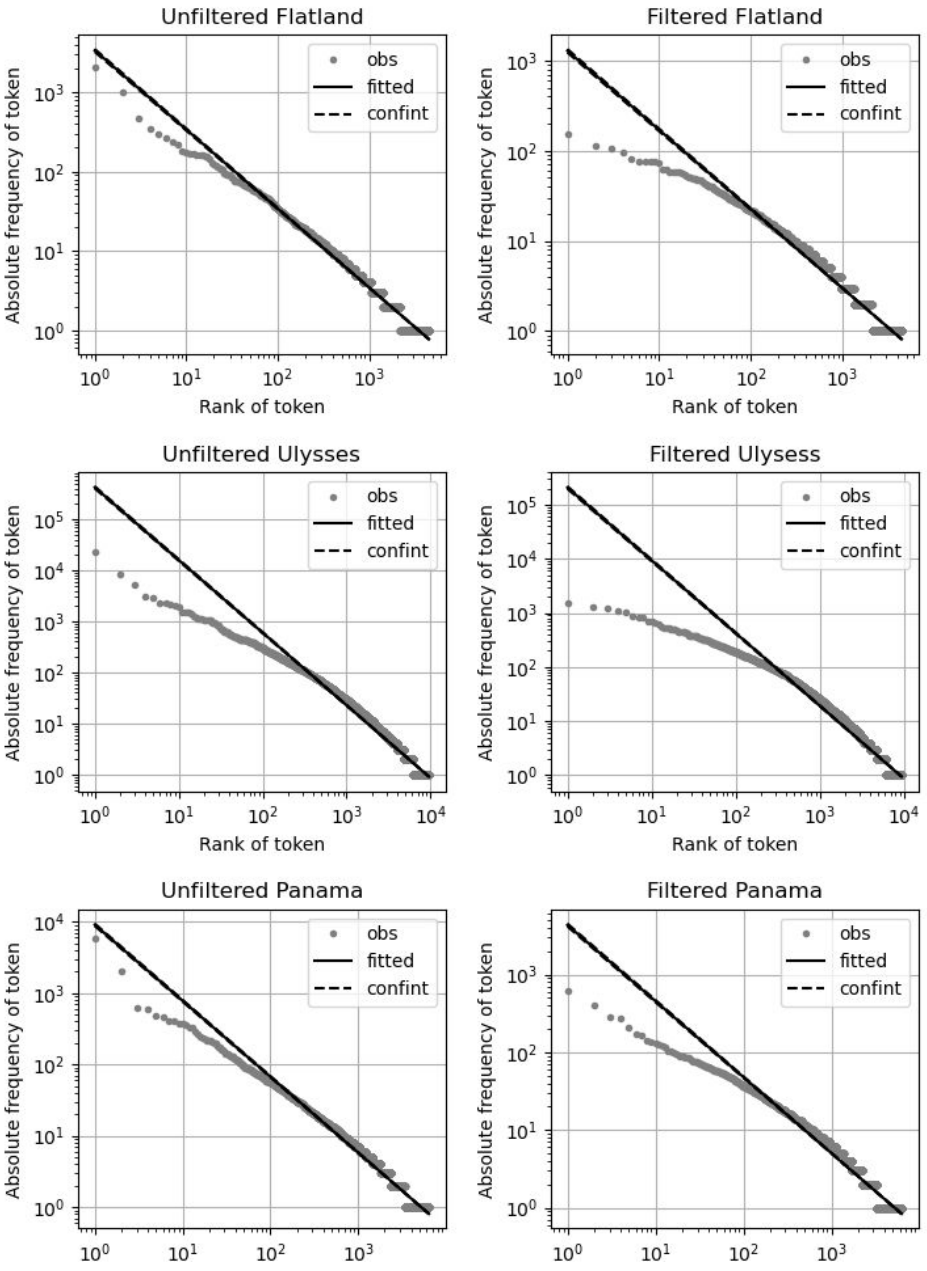

Figure A1 - Tokens for the books plotted in a loglog-plot, overlaid with a linear regression line and the corresponding confidence interval

FIGURE A2 - LOGLOG PLOTS WORDS AND A LINEAR REGRESSION LINE FOR THE CATEGORY-MATCHED TOKENS

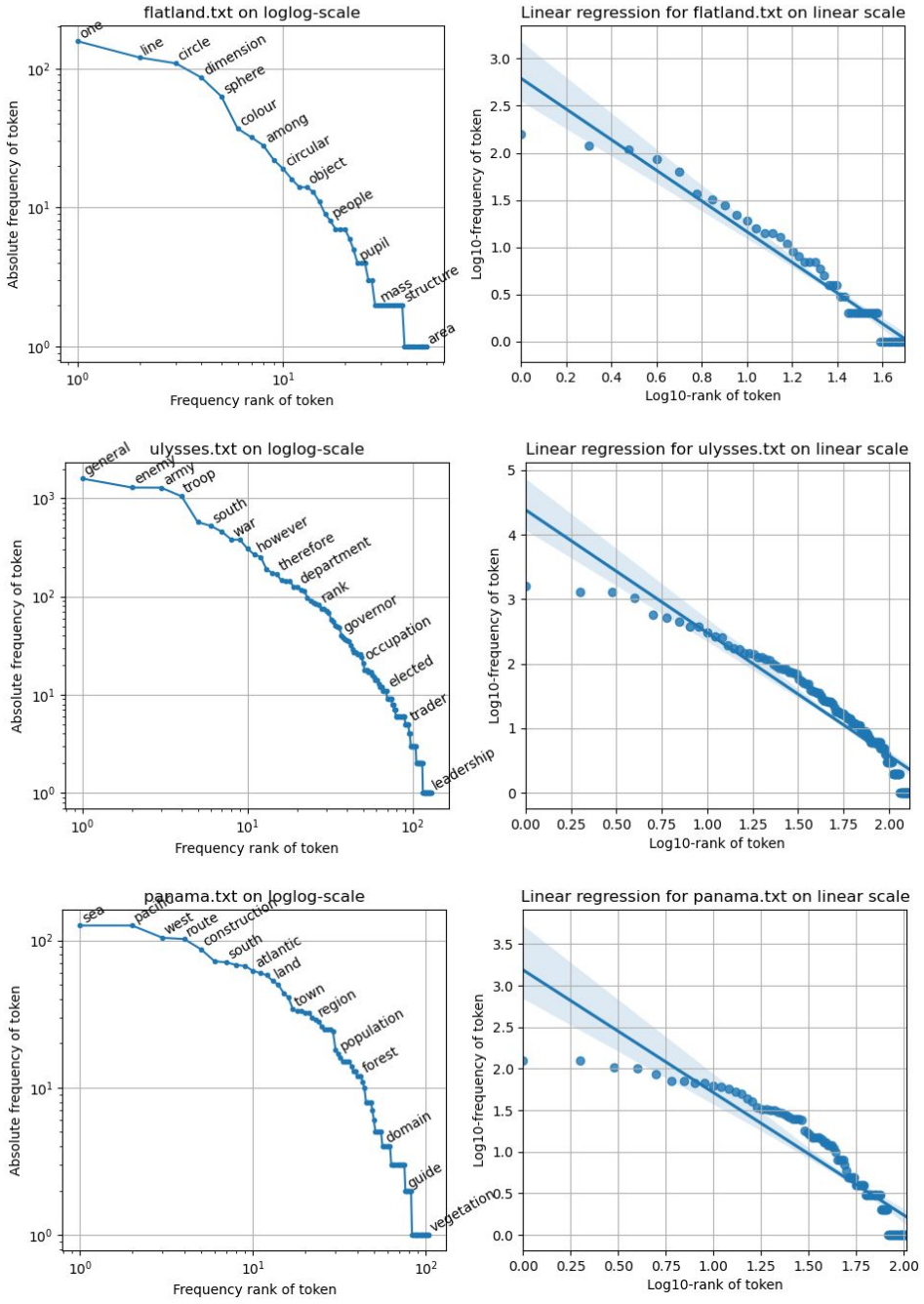

Figure A2 - Loglog-plots for Empath category-matched tokens on the left and linear regression with 95% confidence interval on the right for each book.

Additional script files

[1] Source code for Tasks 1, 2 , 3 , 5 & 8, fitting of linear regression
https://github.com/simppasofia/zipf-law-and-validation/blob/main/appendix/tasks-2-3.py

[2] Jupyter notebook for Task 6, fitting of power law with exponential cutoff,
https://github.com/simppasofia/zipf-law-and-validation/blob/main/appendix/task-6.ipynb

[3] Jupyter notebook for Task 7 (GI part), word categorisation according to General Inquirer categories:
https://github.com/simppasofia/zipf-law-and-validation/blob/main/appendix/task-7.ipynb

[4] Source code for Task 7 (Empath part), word categorisation implemented with Empath tool for python:
https://github.com/simppasofia/zipf-law-and-validation/tree/main/appendix/task-7-empath