In [ ]:

```
# Ahmad Sharif
# DATA.STAT.840-2023-2024-1 Statistical Methods for Text Data Analysis
# K436765
```

In [51]:

```
import requests
import bs4
import urllib.request

from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer
```

In [52]:

```
webpage_url = "https://www.sis.uta.fi/~tojape/"
webpage_html = requests.get(webpage_url)
webpage_parsed_html = bs4.BeautifulSoup(webpage_html.content, 'html.parse
# webpage_parsed_html
```

1. It can crawl the same page multiple times, if a link on a later crawled page points to the already-crawled page.
2. It inserts all links from each page in order as pages to be crawled. If some page contains thousands of links, the crawling will crawl those first and may never get to the links from the next page, especially if the total number of pages are limited. To fix this duplicate issue, at first before insert into the list, I check if the url already exist list data- structure.

In [53]:

```
def getpageurls(webpage_parsed):
    pagelinkelements=webpage_parsed.find_all('a')
    pageurls = [];
    for pagelink in pagelinkelements:
        pageurl_isok=1
        try:
            pageurl = pagelink['href']
        except:
            pageurl_isok=0
        if (pageurl.find('.pdf') !=-1)|(pageurl.find('.ps')!=-1):
            pageurl_isok = 0
        if (pageurl.find('http') ==-1 )|(pageurl.find('.fi')==-1):
            pageurl_isok = 0
        if pageurl_isok == 1 and pageurl not in pageurls: # Before Append
            pageurls.append(pageurl)

    return(pageurls)
mywebpage_urls = getpageurls(webpage_parsed_html)
for x in mywebpage_urls:
    print(x)
# print(mywebpage_urls)
```

https://www.tuni.fi/en

https://www.tuni.fi/en/about-us/faculty-information-technology-and-communication-sciences

https://www.tuni.fi/en/about-us/computing-sciences

http://cs.aalto.fi/en/

http://www.cis.hut.fi/projects/mi

http://users.ics.aalto.fi/jtpelto/

http://research.ics.aalto.fi/coin/

https://www.tuni.fi/en/study-with-us/computing-sciences-data-science?navref=curated--list

https://www.tuni.fi/en/study-with-us/computing-sciences-statistical-data-analytics?navref=curated--list

https://www.tuni.fi/studentsguide/curriculum/degree-programmes/uta-tohjelma-1717?year=2019

https://www.tuni.fi/studentsguide/curriculum/course-units/otm-d42bf3fb-ecd7-43ee-919e-3a18e0b7d885?year=2020&q=null

https://www.tuni.fi/studentsguide/curriculum/course-units/otm-386280c0-c76b-4837-b4e3-61a9d53130b4?year=2020

https://www.tuni.fi/studentsguide/curriculum/course-units/uta-ykoodi-48003?year=2019

https://www.tuni.fi/studentsguide/curriculum/course-units/uta-ykoodi-48010?year=2019

https://www.tuni.fi/studentsguide/curriculum/course-units/uta-ykoodi-38903?year=2019

https://www10.uta.fi/opas/teaching/course.htm?id=32034

https://www10.uta.fi/opas/teaching/course.htm?id=32030

https://www10.uta.fi/opas/opetusohjelma/marjapuuro.htm?id=34169

https://www10.uta.fi/opas/opetusohjelma/marjapuuro.htm?id=32033

https://www10.uta.fi/opas/opetusohjelma/marjapuuro.htm?id=32030

https://www10.uta.fi/opas/opetusohjelma/marjapuuro.htm?id=29901

https://www10.uta.fi/opas/opetusohjelma/marjapuuro.htm?id=29910

https://www10.uta.fi/opas/opetusohjelma/marjapuuro.htm?id=29909

https://www10.uta.fi/opas/opetusohjelma/marjapuuro.htm?id=25061

https://www10.uta.fi/opas/opetusohjelma/marjapuuro.htm?id=24888

https://www10.uta.fi/opas/opetusohjelma/marjapuuro.htm?id=27922

https://www10.uta.fi/opas/opetusohjelma/marjapuuro.htm?id=23239

https://www10.uta.fi/opas/opetusohjelma/marjapuuro.htm?id=20713

https://noppa.aalto.fi/noppa/kurssi/t-61.2020/etusivu

https://noppa.aalto.fi/noppa/kurssi/t-61.5010/etusivu

https://noppa.aalto.fi/noppa/kurssi/t-61.3050/etusivu

https://noppa.aalto.fi/noppa/kurssi/t-61.6040/etusivu

https://noppa.tkk.fi/noppa/kurssi/t-61.6040/etusivu

https://noppa.tkk.fi/noppa/kurssi/t-61.3040/etusivu

https://www.tuni.fi/en/mykola-andrushchenko

http://people.uta.fi/~kauppinen.joonas.t/

https://people.uta.fi/~olli.kuparinen/

http://users.ics.aalto.fi/ziyuang/

http://users.ics.aalto.fi/hani/

http://users.ics.aalto.fi/jstrahl/

https://www.tuni.fi/en/elizaveta-zimina

http://users.ics.aalto.fi/zhexie/

http://users.ics.tkk.fi/faisal/

https://www.tuni.fi/en/joni-pajarinen

http://users.ics.tkk.fi/lgillber/

http://users.ics.aalto.fi/msandhol/

https://www.tuni.fi/en/essi-syrjala

http://users.ics.tkk.fi/kongeo/

http://users.ics.tkk.fi/jviinika/

http://users.ics.aalto.fi/mlosoi/

```
https://journal.fi/sananjalka/article/view/80056
http://www.cis.hut.fi/projects/mi/abstracts/jmlr10.html
http://www.cis.hut.fi/projects/mi/abstracts/ida09b.html
http://www.cis.hut.fi/projects/mi/abstracts/csda09.html
http://www.cis.hut.fi/projects/mi/abstracts/tnn04.html
http://www.cis.hut.fi/projects/mi/abstracts/nn04.html
http://www.cis.hut.fi/projects/mi/abstracts/trnn00.html
http://research.ics.aalto.fi/mi/software/ne/index.shtml
http://www.cis.hut.fi/projects/mi/abstracts/icassp10.html
http://www.cis.hut.fi/projects/mi/abstracts/wsom09.html
http://www.cis.hut.fi/projects/mi/abstracts/icassp09_snerv.html
http://www.cis.hut.fi/projects/mi/abstracts/ecml07.html
http://www.cis.hut.fi/projects/mi/abstracts/mlsp07.html
http://www.cis.hut.fi/projects/mi/abstracts/pmsb2006.html
http://www.cis.hut.fi/projects/mi/abstracts/icml04.html
http://www.cis.hut.fi/projects/mi/abstracts/wsom03b.html
http://www.cis.hut.fi/projects/mi/abstracts/icml03.html
http://www.cis.hut.fi/projects/mi/abstracts/ecml03.html
http://www.cis.hut.fi/projects/mi/abstracts/iconip02.html
http://www.cis.hut.fi/projects/mi/abstracts/icann02.html
http://www.cis.hut.fi/projects/mi/abstracts/ijcnn01.html
https://politiikasta.fi/parlamentaarisen-politiikan-ajat/
http://www.cis.hut.fi/projects/mi/abstracts/nips08_lms_rsl.html
http://www.nbl.fi/~nbl924/renkula/
http://www.cis.hut.fi/projects/mi/abstracts/nips06_did.html
http://www.cis.hut.fi/projects/mi/abstracts/nips06_lce.html
http://www.cis.hut.fi/projects/mi/abstracts/eccb06poster.html
http://lib.hut.fi/Diss/2004/isbn9512273454/
```

In [54]:

```python
webpage_url = "https://gutenberg.org/browse/scores/top"
webpage_html = requests.get(webpage_url)
webpage_parsed_html = bs4.BeautifulSoup(webpage_html.content, 'html.parse
```

2) a)

In [55]:

```python
pageList = webpage_parsed_html.find("h2", {"id": "books-last30"})
ol = pageList.find_next_sibling("ol")
book_list = {}


def collect_all_download_link(count):
    index = 0
    for x in (ol.findAll('li')):
        index = index + 1
        url = x.a['href']

        book_id = url.split('/')[2]
        book_name = x.a.text
        download_link = 'https://www.gutenberg.org/files/' + book_id + '/
        print(index, " : ", download_link)
        if(index < count):
            book_list[book_id] = {
                "book_name": book_name,
                "download_link": download_link
            }
collect_all_download_link(20)
```

```
print(len(book_list))
# 20
```

```
1  :  https://www.gutenberg.org/files/84/84-0.txt
2  :  https://www.gutenberg.org/files/1513/1513-0.txt
3  :  https://www.gutenberg.org/files/1342/1342-0.txt
4  :  https://www.gutenberg.org/files/25344/25344-0.txt
5  :  https://www.gutenberg.org/files/11/11-0.txt
6  :  https://www.gutenberg.org/files/345/345-0.txt
7  :  https://www.gutenberg.org/files/174/174-0.txt
8  :  https://www.gutenberg.org/files/5200/5200-0.txt
9  :  https://www.gutenberg.org/files/64317/64317-0.txt
10  :  https://www.gutenberg.org/files/2542/2542-0.txt
11  :  https://www.gutenberg.org/files/1952/1952-0.txt
12  :  https://www.gutenberg.org/files/1080/1080-0.txt
13  :  https://www.gutenberg.org/files/2701/2701-0.txt
14  :  https://www.gutenberg.org/files/844/844-0.txt
15  :  https://www.gutenberg.org/files/43/43-0.txt
16  :  https://www.gutenberg.org/files/98/98-0.txt
17  :  https://www.gutenberg.org/files/41/41-0.txt
18  :  https://www.gutenberg.org/files/1661/1661-0.txt
19  :  https://www.gutenberg.org/files/6130/6130-0.txt
20  :  https://www.gutenberg.org/files/408/408-0.txt
21  :  https://www.gutenberg.org/files/1232/1232-0.txt
22  :  https://www.gutenberg.org/files/1260/1260-0.txt
23  :  https://www.gutenberg.org/files/1400/1400-0.txt
24  :  https://www.gutenberg.org/files/28054/28054-0.txt
25  :  https://www.gutenberg.org/files/2591/2591-0.txt
26  :  https://www.gutenberg.org/files/1399/1399-0.txt
27  :  https://www.gutenberg.org/files/26184/26184-0.txt
28  :  https://www.gutenberg.org/files/46/46-0.txt
29  :  https://www.gutenberg.org/files/76/76-0.txt
30  :  https://www.gutenberg.org/files/2554/2554-0.txt
31  :  https://www.gutenberg.org/files/16328/16328-0.txt
32  :  https://www.gutenberg.org/files/3207/3207-0.txt
33  :  https://www.gutenberg.org/files/219/219-0.txt
34  :  https://www.gutenberg.org/files/4300/4300-0.txt
35  :  https://www.gutenberg.org/files/205/205-0.txt
36  :  https://www.gutenberg.org/files/2814/2814-0.txt
37  :  https://www.gutenberg.org/files/16/16-0.txt
38  :  https://www.gutenberg.org/files/7370/7370-0.txt
39  :  https://www.gutenberg.org/files/23/23-0.txt
40  :  https://www.gutenberg.org/files/1184/1184-0.txt
41  :  https://www.gutenberg.org/files/1727/1727-0.txt
42  :  https://www.gutenberg.org/files/27827/27827-0.txt
43  :  https://www.gutenberg.org/files/2600/2600-0.txt
44  :  https://www.gutenberg.org/files/1497/1497-0.txt
45  :  https://www.gutenberg.org/files/41445/41445-0.txt
46  :  https://www.gutenberg.org/files/74/74-0.txt
47  :  https://www.gutenberg.org/files/15399/15399-0.txt
48  :  https://www.gutenberg.org/files/10007/10007-0.txt
49  :  https://www.gutenberg.org/files/55/55-0.txt
50  :  https://www.gutenberg.org/files/768/768-0.txt
51  :  https://www.gutenberg.org/files/5740/5740-0.txt
52  :  https://www.gutenberg.org/files/45/45-0.txt
53  :  https://www.gutenberg.org/files/932/932-0.txt
54  :  https://www.gutenberg.org/files/2000/2000-0.txt
55  :  https://www.gutenberg.org/files/58585/58585-0.txt
56  :  https://www.gutenberg.org/files/996/996-0.txt
57  :  https://www.gutenberg.org/files/1998/1998-0.txt
58  :  https://www.gutenberg.org/files/2148/2148-0.txt
59  :  https://www.gutenberg.org/files/244/244-0.txt
```

```
60  :  https://www.gutenberg.org/files/67098/67098-0.txt
61  :  https://www.gutenberg.org/files/120/120-0.txt
62  :  https://www.gutenberg.org/files/2680/2680-0.txt
63  :  https://www.gutenberg.org/files/33283/33283-0.txt
64  :  https://www.gutenberg.org/files/2852/2852-0.txt
65  :  https://www.gutenberg.org/files/8800/8800-0.txt
66  :  https://www.gutenberg.org/files/514/514-0.txt
67  :  https://www.gutenberg.org/files/20203/20203-0.txt
68  :  https://www.gutenberg.org/files/30254/30254-0.txt
69  :  https://www.gutenberg.org/files/6133/6133-0.txt
70  :  https://www.gutenberg.org/files/4363/4363-0.txt
71  :  https://www.gutenberg.org/files/600/600-0.txt
72  :  https://www.gutenberg.org/files/42324/42324-0.txt
73  :  https://www.gutenberg.org/files/38065/38065-0.txt
74  :  https://www.gutenberg.org/files/35/35-0.txt
75  :  https://www.gutenberg.org/files/158/158-0.txt
76  :  https://www.gutenberg.org/files/36/36-0.txt
77  :  https://www.gutenberg.org/files/779/779-0.txt
78  :  https://www.gutenberg.org/files/5827/5827-0.txt
79  :  https://www.gutenberg.org/files/521/521-0.txt
80  :  https://www.gutenberg.org/files/161/161-0.txt
81  :  https://www.gutenberg.org/files/3825/3825-0.txt
82  :  https://www.gutenberg.org/files/10/10-0.txt
83  :  https://www.gutenberg.org/files/8492/8492-0.txt
84  :  https://www.gutenberg.org/files/3296/3296-0.txt
85  :  https://www.gutenberg.org/files/1250/1250-0.txt
86  :  https://www.gutenberg.org/files/147/147-0.txt
87  :  https://www.gutenberg.org/files/44956/44956-0.txt
88  :  https://www.gutenberg.org/files/31284/31284-0.txt
89  :  https://www.gutenberg.org/files/11030/11030-0.txt
90  :  https://www.gutenberg.org/files/35899/35899-0.txt
91  :  https://www.gutenberg.org/files/730/730-0.txt
92  :  https://www.gutenberg.org/files/100/100-0.txt
93  :  https://www.gutenberg.org/files/203/203-0.txt
94  :  https://www.gutenberg.org/files/209/209-0.txt
95  :  https://www.gutenberg.org/files/236/236-0.txt
96  :  https://www.gutenberg.org/files/24869/24869-0.txt
97  :  https://www.gutenberg.org/files/4217/4217-0.txt
98  :  https://www.gutenberg.org/files/1251/1251-0.txt
99  :  https://www.gutenberg.org/files/851/851-0.txt
100  :  https://www.gutenberg.org/files/6753/6753-0.txt
19
```

In [6]:

```python
## 2 > b Name And Link
index = 0
for x in book_list:
    index = index + 1
    print(index, "Book Name     :", book_list[x]["book_name"])
    print("  Download Link :", book_list[x]["download_link"])
    print()
```

```
1 Book Name     : Frankenstein; Or, The Modern Prometheus by Mary Wollston
ecraft Shelley (83654)
  Download Link : https://www.gutenberg.org/files/84/84-0.txt

2 Book Name     : Romeo and Juliet by William Shakespeare (63125)
  Download Link : https://www.gutenberg.org/files/1513/1513-0.txt

3 Book Name     : Pride and Prejudice by Jane Austen (54666)
  Download Link : https://www.gutenberg.org/files/1342/1342-0.txt

4 Book Name     : The Scarlet Letter by Nathaniel Hawthorne (37534)
  Download Link : https://www.gutenberg.org/files/25344/25344-0.txt

5 Book Name     : Alice's Adventures in Wonderland by Lewis Carroll (3051
7)
  Download Link : https://www.gutenberg.org/files/11/11-0.txt

6 Book Name     : Dracula by Bram Stoker (29837)
  Download Link : https://www.gutenberg.org/files/345/345-0.txt

7 Book Name     : The Picture of Dorian Gray by Oscar Wilde (25794)
  Download Link : https://www.gutenberg.org/files/174/174-0.txt

8 Book Name     : Metamorphosis by Franz Kafka (24788)
  Download Link : https://www.gutenberg.org/files/5200/5200-0.txt

9 Book Name     : The Great Gatsby by F. Scott  Fitzgerald (24469)
  Download Link : https://www.gutenberg.org/files/64317/64317-0.txt

10 Book Name     : A Doll's House : a play by Henrik Ibsen (22998)
  Download Link : https://www.gutenberg.org/files/2542/2542-0.txt

11 Book Name     : The Yellow Wallpaper by Charlotte Perkins Gilman (2242
6)
  Download Link : https://www.gutenberg.org/files/1952/1952-0.txt

12 Book Name     : A Modest Proposal by Jonathan Swift (21398)
  Download Link : https://www.gutenberg.org/files/1080/1080-0.txt

13 Book Name     : Moby Dick; Or, The Whale by Herman Melville (19991)
  Download Link : https://www.gutenberg.org/files/2701/2701-0.txt

14 Book Name     : The Importance of Being Earnest: A Trivial Comedy for S
erious People by Oscar Wilde (19579)
  Download Link : https://www.gutenberg.org/files/844/844-0.txt

15 Book Name     : The Strange Case of Dr. Jekyll and Mr. Hyde by Robert L
ouis Stevenson (19402)
  Download Link : https://www.gutenberg.org/files/43/43-0.txt

16 Book Name     : A Tale of Two Cities by Charles Dickens (18363)
  Download Link : https://www.gutenberg.org/files/98/98-0.txt

17 Book Name     : The Legend of Sleepy Hollow by Washington Irving (1723
4)
  Download Link : https://www.gutenberg.org/files/41/41-0.txt

18 Book Name     : The Adventures of Sherlock Holmes by Arthur Conan Doyle
(16994)
```

```
  Download Link : https://www.gutenberg.org/files/1661/1661-0.txt

19 Book Name     : The Iliad by Homer (16269)
  Download Link : https://www.gutenberg.org/files/6130/6130-0.txt
```

In [50]:

```
# 2 ) c) Use the processing pipeline described on the lecture to tokenize

import nltk
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer
import urllib


crawled_lemmatize_word_list = []
book_id = list(book_list.keys())[0]
url = "https://www.gutenberg.org/files/" + book_id + "/" + book_id + "-0.
book_content = urllib.request.urlopen(url).read().decode('utf-8')

# print(book_content)



# Tokenize
book_word_list = word_tokenize(book_content)
# print(book_word_list)


# Lemmatize
lemmatizer = WordNetLemmatizer()
for word in book_word_list:
    l = lemmatizer.lemmatize(word).lower()
    nltk_text = nltk.Text(l)
    crawled_lemmatize_word_list.append(l)
```

In [29]:

```
'''for x in crawled_lemmatize_word_list:
    print(x)'''
```

Out[29]:

```
'for x in crawled_lemmatize_word_list:\n    print(x)'
```

In [36]:

```
# 2.2 s) (d) Create a unified vocabulary from the ebooks; report the top-

import numpy as np
from nltk.probability import FreqDist

r = np.unique(crawled_lemmatize_word_list, return_inverse=True)

print(r[0][130:230])


vocabulary = sorted(set(r[0]))
# print(vocab)
```

```
['abandoned' 'abbey' 'abhor' 'abhorred' 'abhorred.' 'abhorrence'
 'abhorrent' 'abide' 'ability' 'abject' 'able' 'aboard' 'abode' 'abortion'
 'abortive' 'about' 'above' 'abroad' 'abrupt' 'absence' 'absence.'
 'absent' 'absolute' 'absolutely' 'absolution' 'absorbed' 'absorbing'
 'abstained' 'abstruse' 'abyss' 'acceded' 'accent' 'accept' 'acceptance'
 'accepted' 'accepting' 'access' 'accessed' 'accessible' 'accident'
 'accidentally' 'accompanied' 'accompany' 'accomplish' 'accomplished'
 'accomplishment' 'accordance' 'accorded' 'according' 'accordingly'
 'account' 'accounted' 'accumulated' 'accumulation' 'accuracy' 'accurate'
 'accursed' 'accusation' 'accuse' 'accused' 'accuses' 'accustomed'
 'achieve' 'achieved' 'achievement' 'aching' 'acknowledged' 'acorn'
 'acquaintance' 'acquainted' 'acquiesced' 'acquire' 'acquired'
 'acquirement' 'acquiring' 'acquisition' 'acquit' 'acquittal.' 'acquitted'
 'acquitted.' 'across' 'act' 'acted' 'acting' 'action' 'active' 'activity'
 'actor' 'actual' 'actually' 'actuated' 'act–i' 'acuteness' 'adam' 'adapt'
 'add' 'added' 'addition' 'additional' 'address']
```

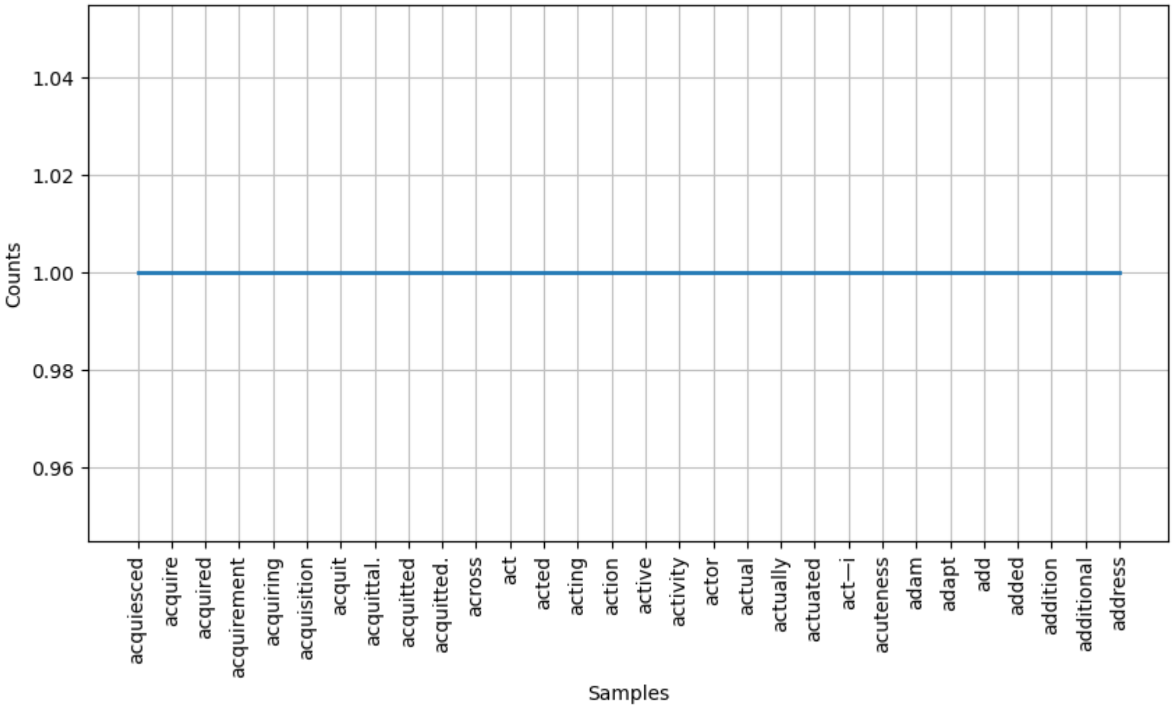
In [41]:

```python
# Exercise 2.3: Zipf's law.
""" Use the top-20 books from Project Gutenberg to examine whether Zipf's
(a) Compute a plot of the frequencies of all words in the vocabulary (= c
divided by the total count of all words together), sorted by frequency. R

import matplotlib.pyplot as plt
from nltk.probability import FreqDist

freq_dist = FreqDist(vocabulary[200:230])
sorted_freq = sorted(freq_dist.items(), key=lambda item: item[1], reverse

# Create a plot
plt.figure(figsize=(10, 5))
freq_dist.plot(30, cumulative=False)
plt.show()
```

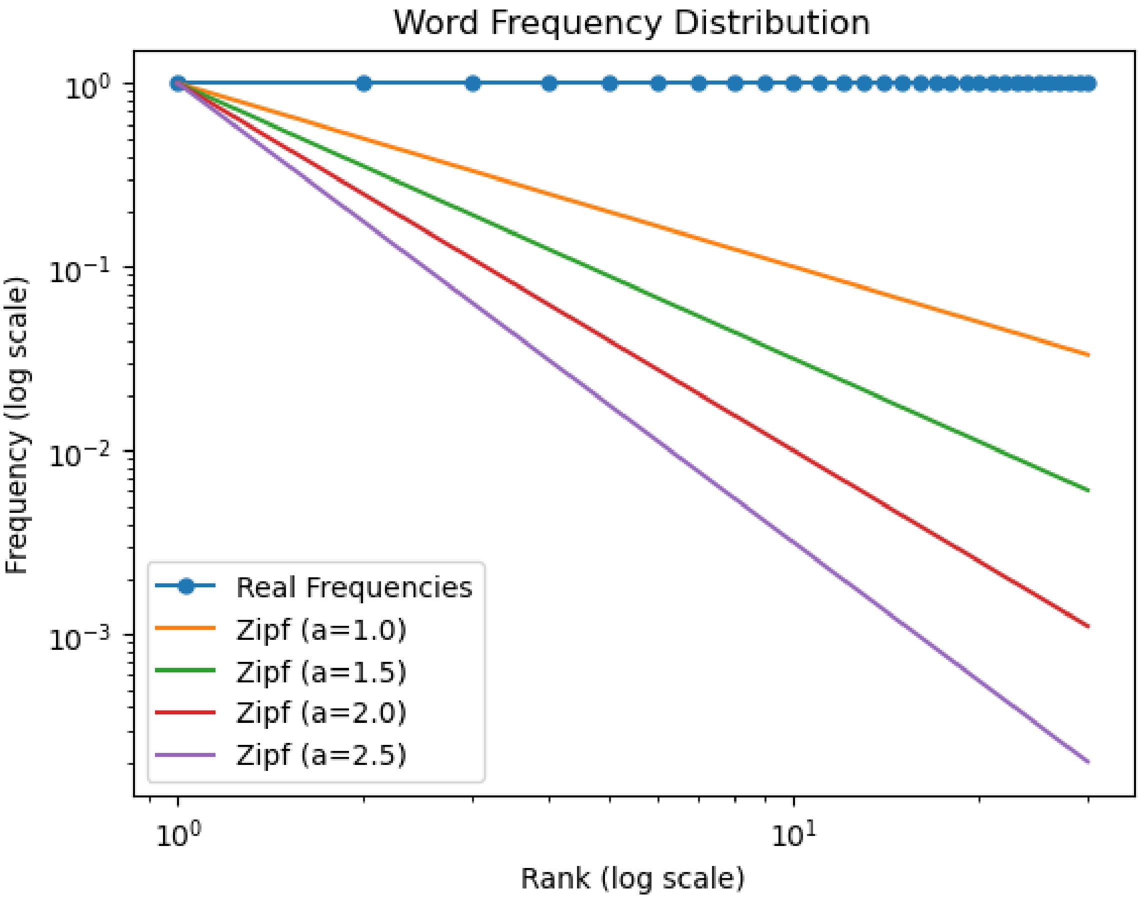


In [42]:

```python
x = np.arange(1, len(sorted_freq) + 1)
y_real = [freq for word, freq in sorted_freq]
plt.plot(x, y_real, label="Real Frequencies", marker='o', markersize=5)
```

```
# Try different values of the exponent 'a'
for a in [1.0, 1.5, 2.0, 2.5]:
    y_zipf = [1 / (rank ** a) for rank in x]
    plt.plot(x, y_zipf, label=f'Zipf (a={a})')

plt.yscale('log')
plt.xscale('log')
plt.xlabel('Rank (log scale)')
plt.ylabel('Frequency (log scale)')
plt.legend()
plt.title("Word Frequency Distribution")
plt.show()
```



Exercise 2.4: Why is Zipf's law important? The lectures introduced Zipf's law, but why is it important? (a) Ask an AI chatbot what the importance of Zipf's law is; report what chatbot you used, your query, and the chatbot's answer. (If you use several queries and answers, report them all.)

(b) Improve the chatbot's answer - rewrite it to add more insights. Try to provide at least three more insights that Zipf's law could provide. Report your modified answer with the modified parts of text highlighted.

2. a) In short, Zipf's law is important because it provides a mathematical model for the uneven distribution of elements in various datasets, from word frequencies in language to income distribution in economics. It's used in linguistics, information theory, data analysis, recommendation systems, and more to understand and model real-world phenomena with unequal distributions.

From https://chat.openai.com/

b)

1. It gives an mathematical model of word distribution and frequencies.
2. In addition, this laws also applicable in population size, city size, birth rate, death, online usage and many more