Lesson2: Modelling the Web with Simple Statistical Descriptive Text Models

Unit4: Test if lesser words are required on Simple English Wikipedia to understand a larger fraction than on English Wikipedia

Rene Pickhardt

Introduction to Web Science Part 2

Emerging Web Properties

WeST

People and Knowledge Networks

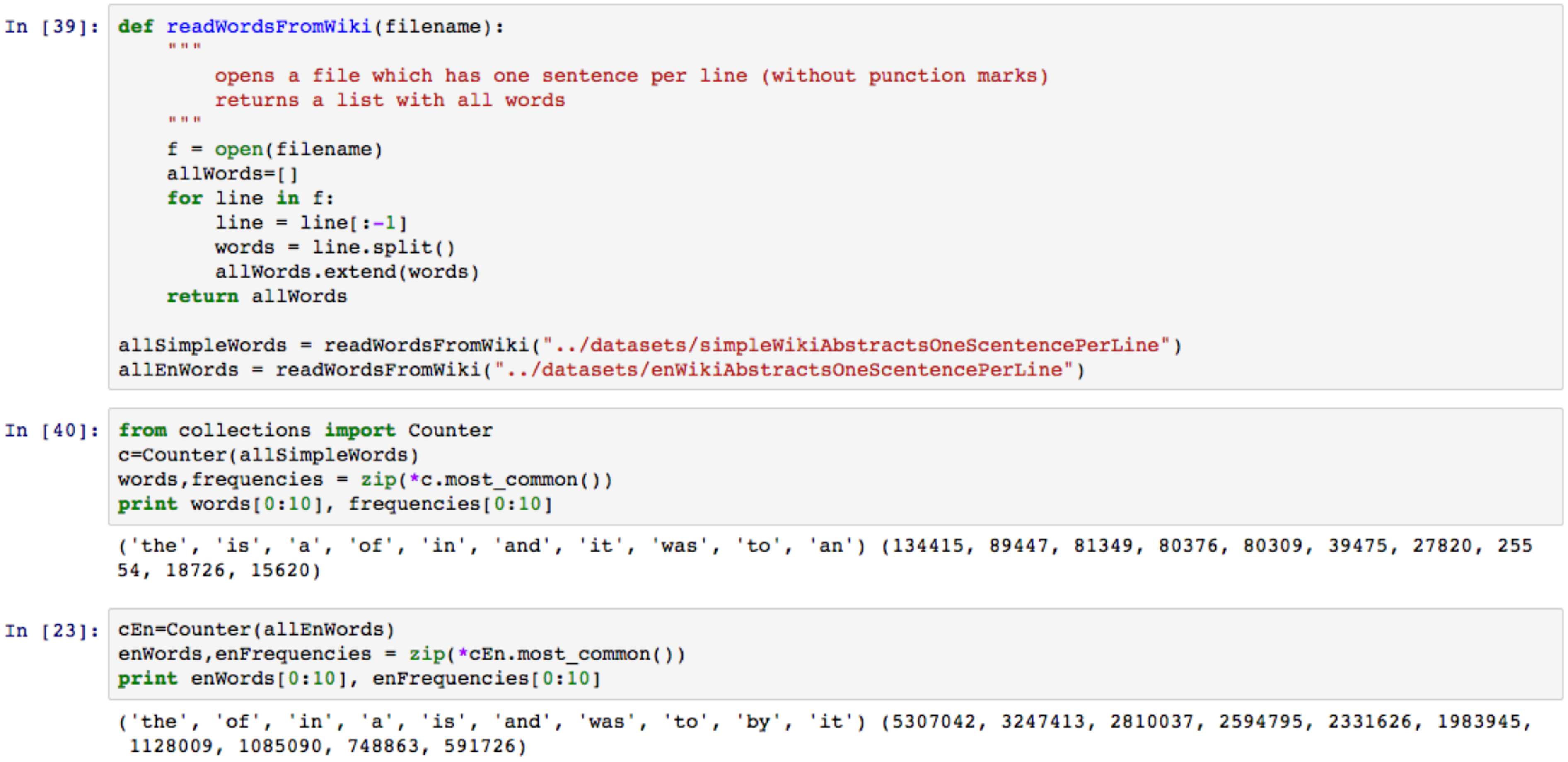
Completing this unit you should

- Understand what a log-log plot is
- Improve your skills in reading and interpreting diagrams
- Know about the word rank / frequency plot
- Should be able to transfer a histogram or curve into a cumulative distribution function

Strategy to fulfil our test

- count the frequency of words in both corpora
- Sort the words descending to their frequency
  - This creates a ranking
- Create a plot displaying the frequency depending on the rank
- Transform this to the cumulative plot in order to test our prediction

Counting words is really simple in python



```
In [39]: def readWordsFromWiki(filename):
             """
                 opens a file which has one sentence per line (without punction marks)
                 returns a list with all words
             """
             f = open(filename)
             allWords=[]
             for line in f:
                 line = line[:-1]
                 words = line.split()
                 allWords.extend(words)
             return allWords

         allSimpleWords = readWordsFromWiki("../datasets/simpleWikiAbstractsOneScentencePerLine")
         allEnWords = readWordsFromWiki("../datasets/enWikiAbstractsOneScentencePerLine")
```

```
In [40]: from collections import Counter
         c=Counter(allSimpleWords)
         words,frequencies = zip(*c.most_common())
         print words[0:10], frequencies[0:10]
```

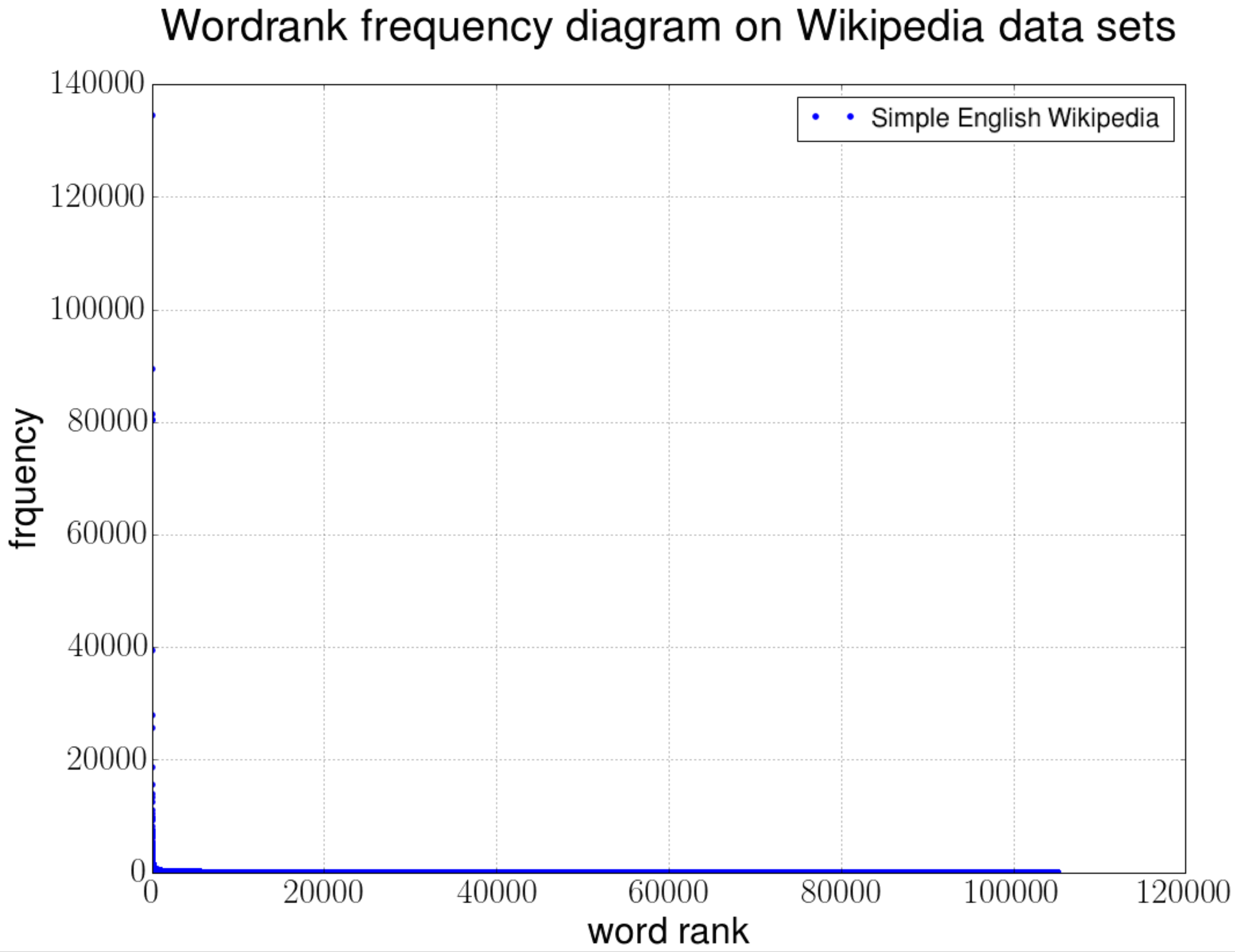
```
('the', 'is', 'a', 'of', 'in', 'and', 'it', 'was', 'to', 'an') (134415, 89447, 81349, 80376, 80309, 39475, 27820, 255
54, 18726, 15620)
```

```
In [23]: cEn=Counter(allEnWords)
         enWords,enFrequencies = zip(*cEn.most_common())
         print enWords[0:10], enFrequencies[0:10]
```

```
('the', 'of', 'in', 'a', 'is', 'and', 'was', 'to', 'by', 'it') (5307042, 3247413, 2810037, 2594795, 2331626, 1983945,
 1128009, 1085090, 748863, 591726)
```

Lets look at the rank frequency diagram

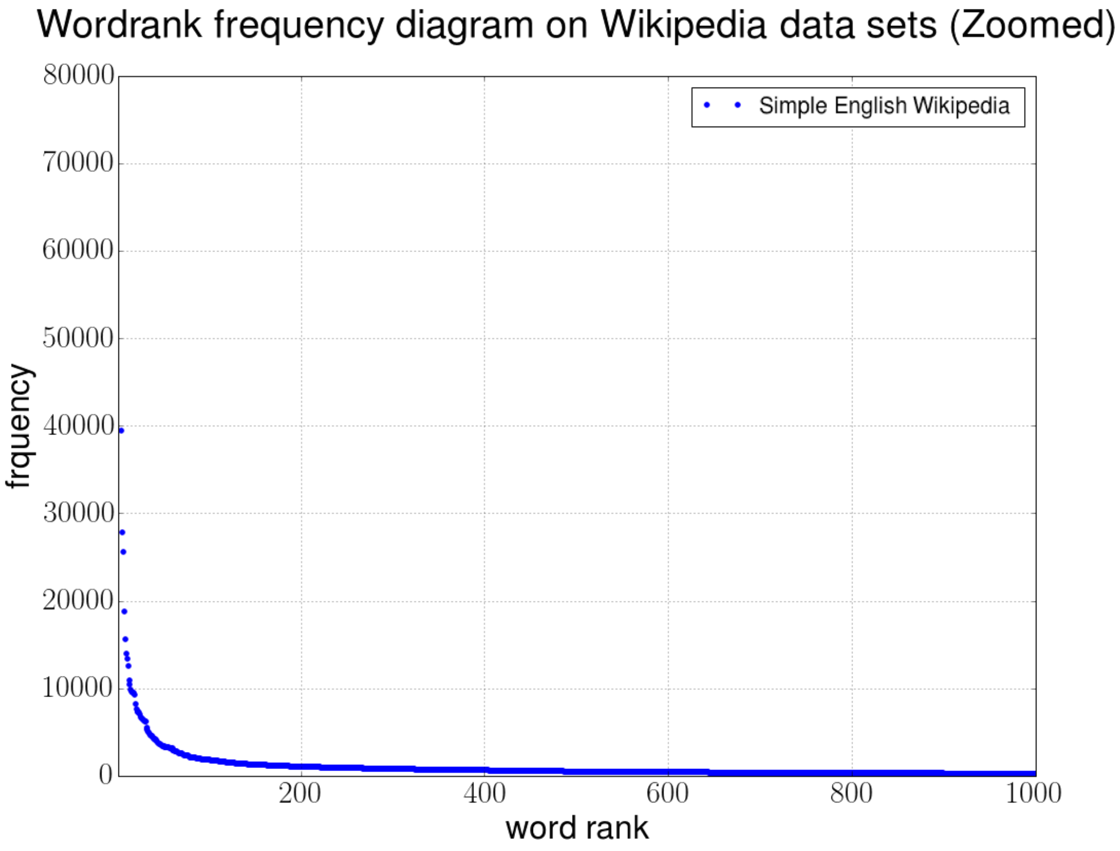
- As you can see, you see nothing (:



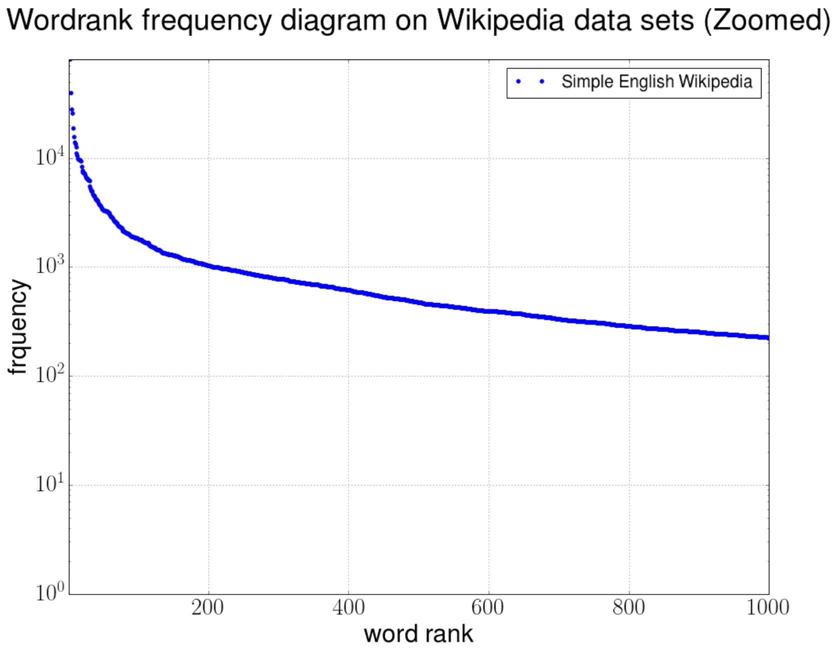
Maybe zooming the axis helps a little bit?

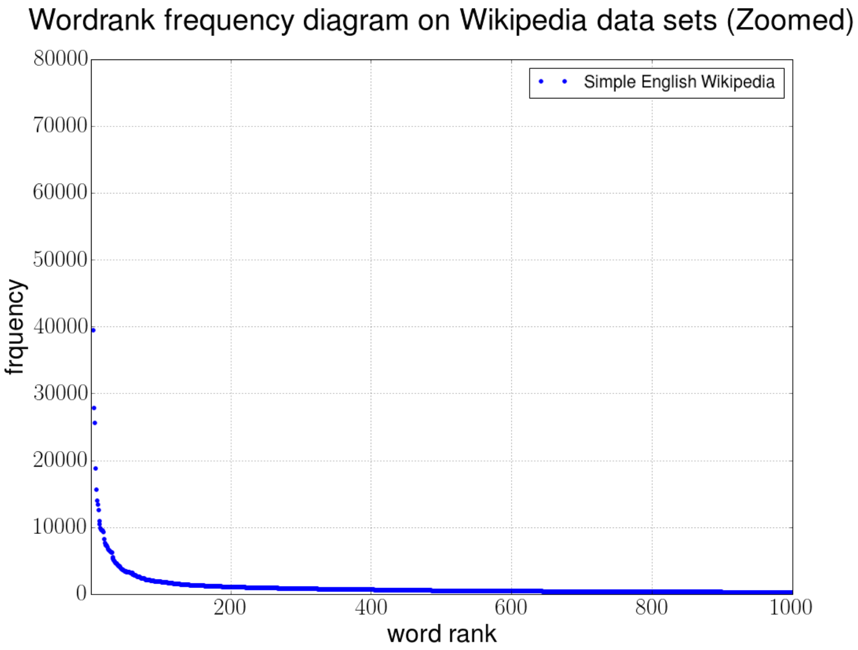
Problems with this plot:

- The frequency of words with a rank bigger than 200 can not be distinguished
- What if all ranks should be displayed?



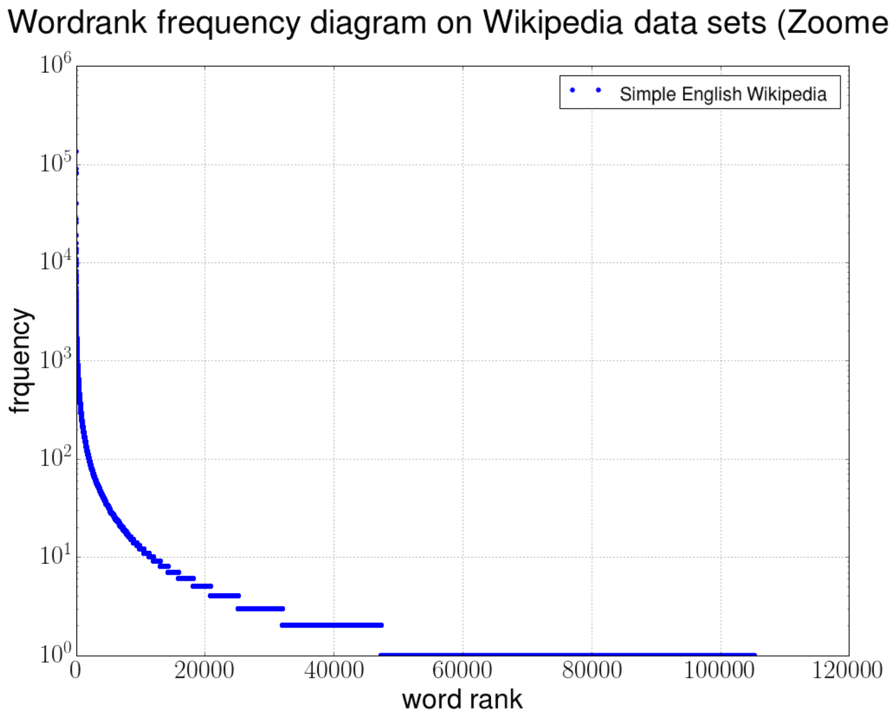
Changing the y-axis to a logarithmic scale





- Same data being used
- Very different visualization
- What happens if we include all ranks again?

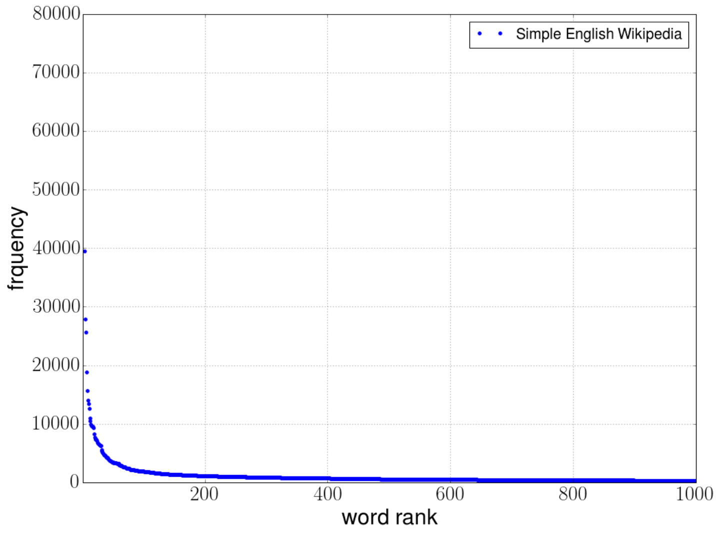
Displaying the full x-axis



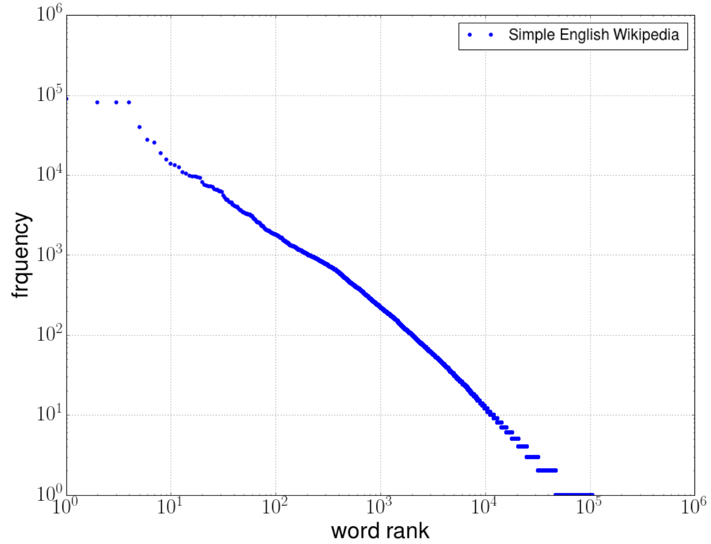
- Similar problems as before:
  - Top ranks can almost not be distinguished
- Do the same trick as before

Compare linear scale plot with log-log plot

Wordrank frequency diagram on Wikipedia data sets (Zoomed)

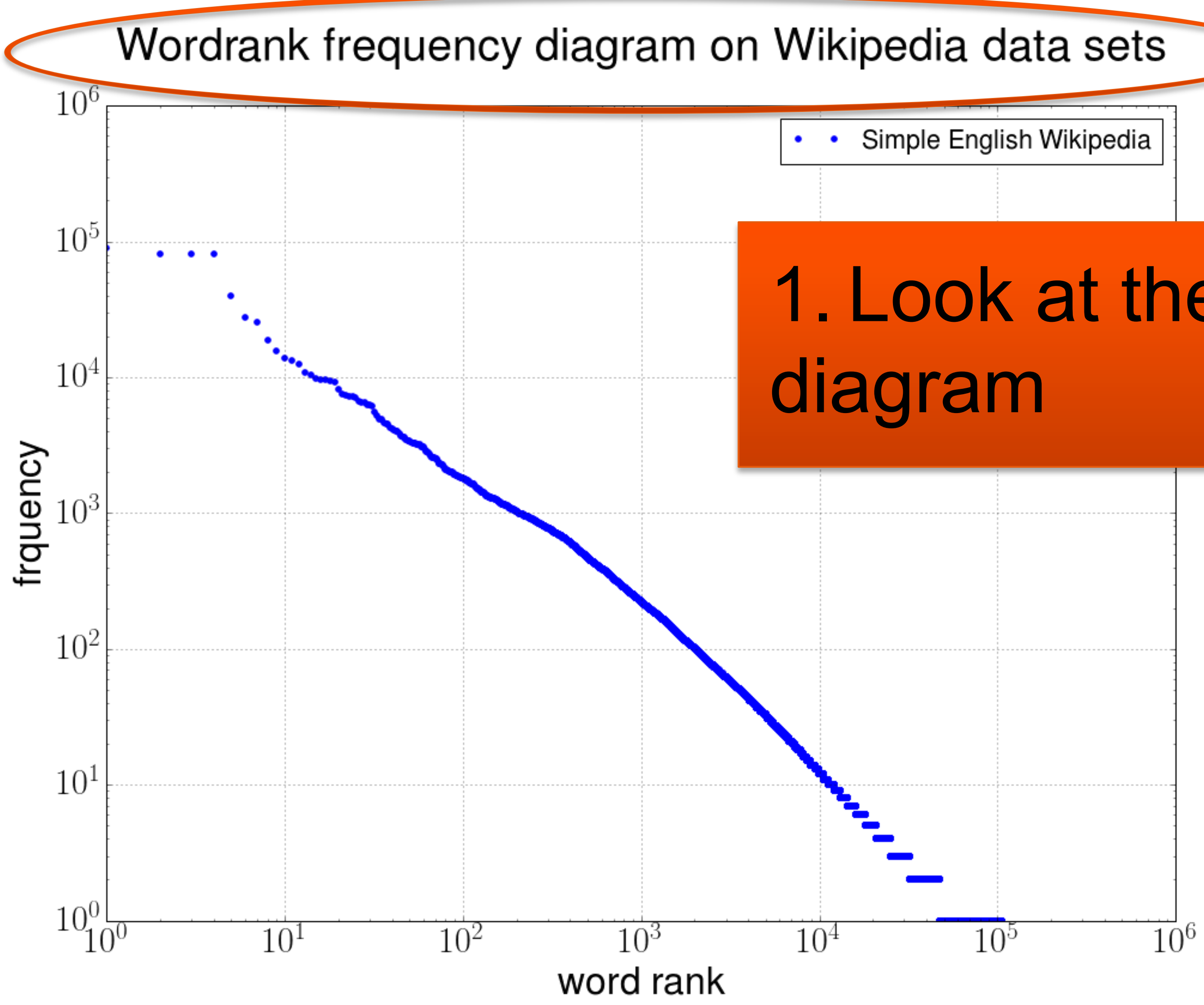


Wordrank frequency diagram on Wikipedia data sets (log-log scale)



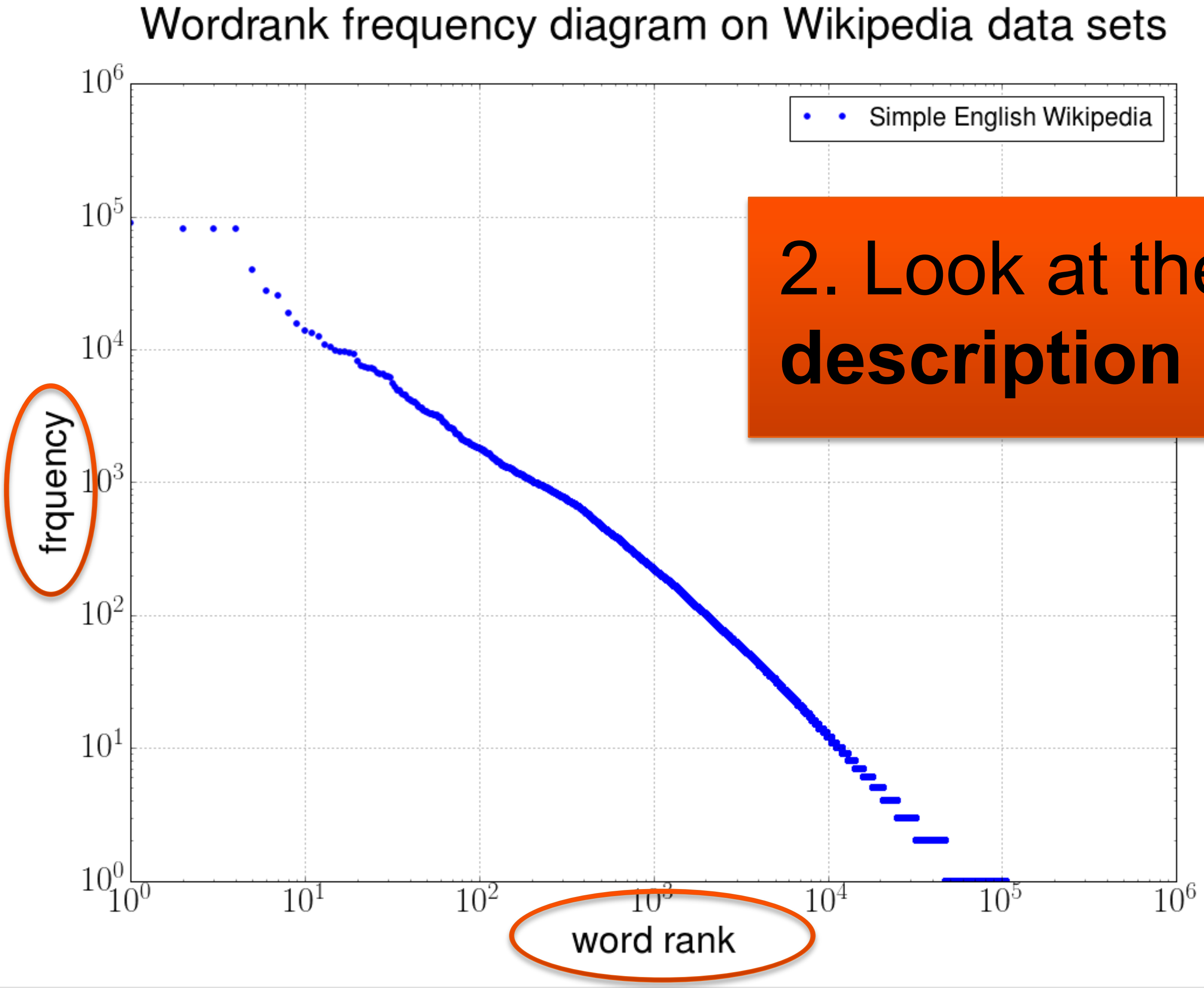
Linear	Logarithmic
Every interval displays a **fixed range** of numbers	Every interval displays one **order of magnitude**
Adding a constant number (10 k) to go from one scale unit to the next one	**Multiplying** with a constant number (in our case 10) to go from one scale to the next
Can visualize best what is happening **in a certain interval** - Usually the highest order of magnitude	Can visualize best what happens in **each** order of magnitude

6 Steps of mastering reading (log-log) plots



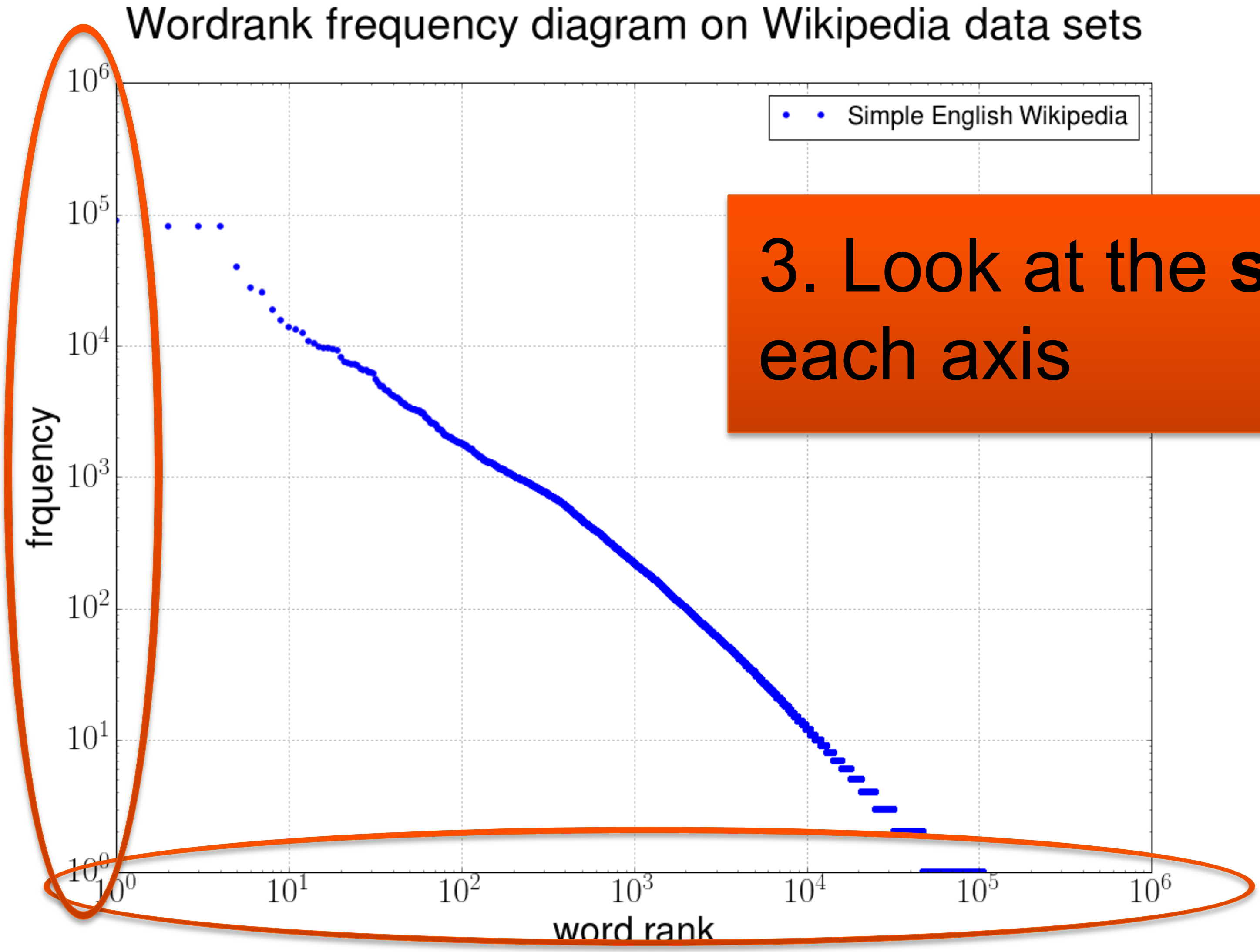
1. Look at the **title** of the diagram

6 Steps of mastering reading (log-log) plots



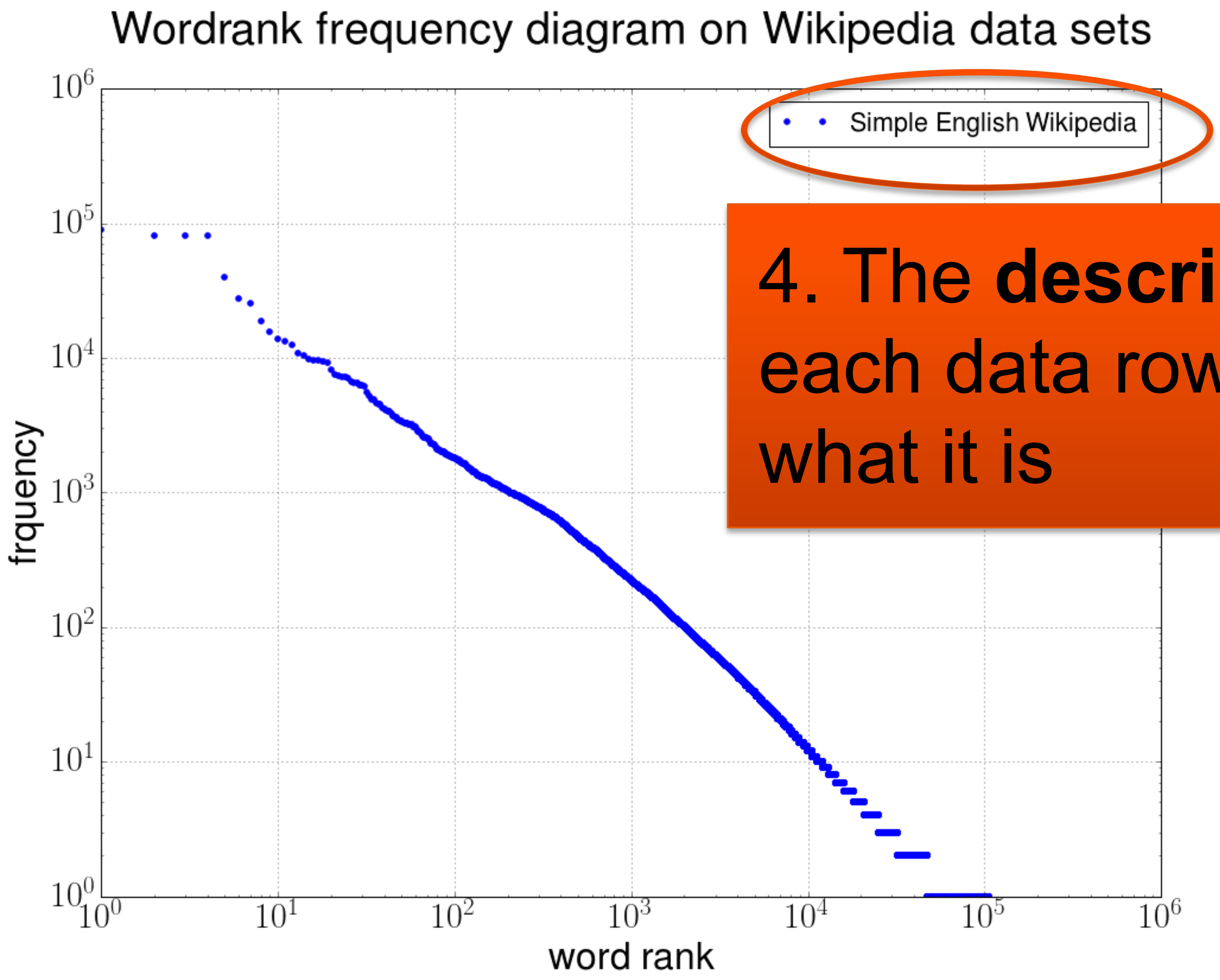
2. Look at the **description** of both axis

6 Steps of mastering reading (log-log) plots



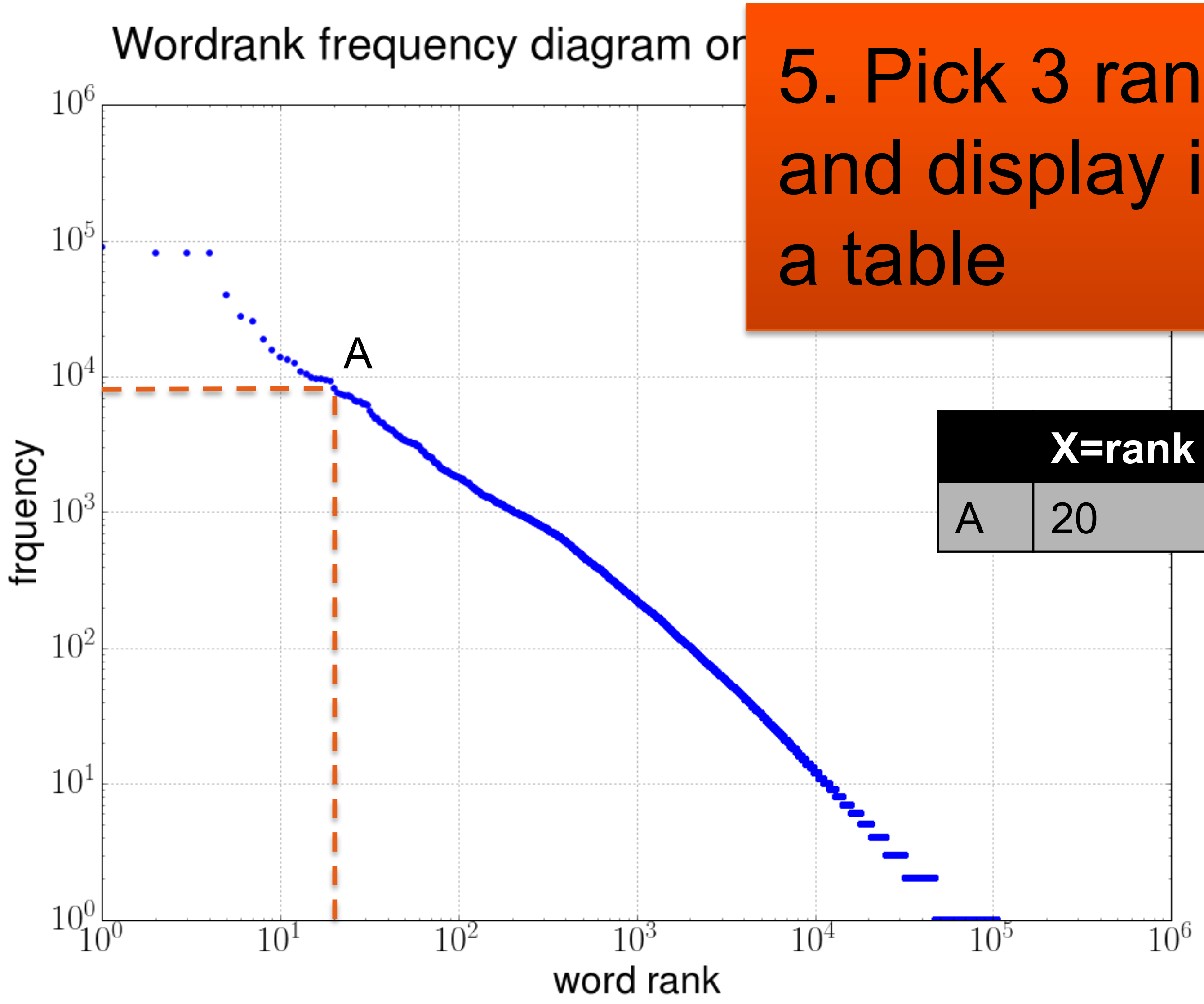
3. Look at the **scale** of each axis

6 Steps of mastering reading (log-log) plots



4. The **description** of each data row tells you what it is

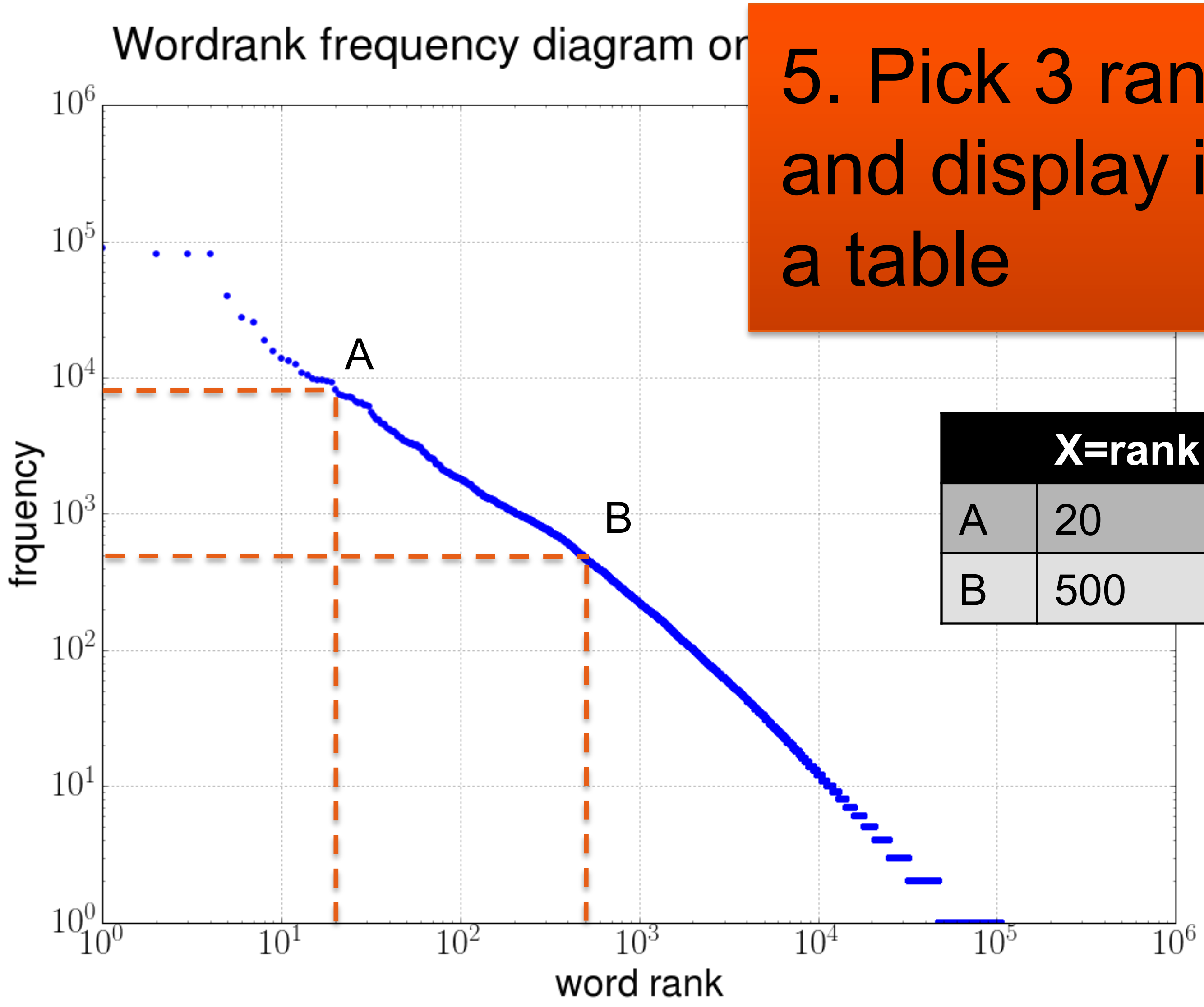
6 Steps of mastering reading (log-log) plots



5. Pick 3 random points and display its values in a table

	X=rank	Y=frequency
A	20	~8000

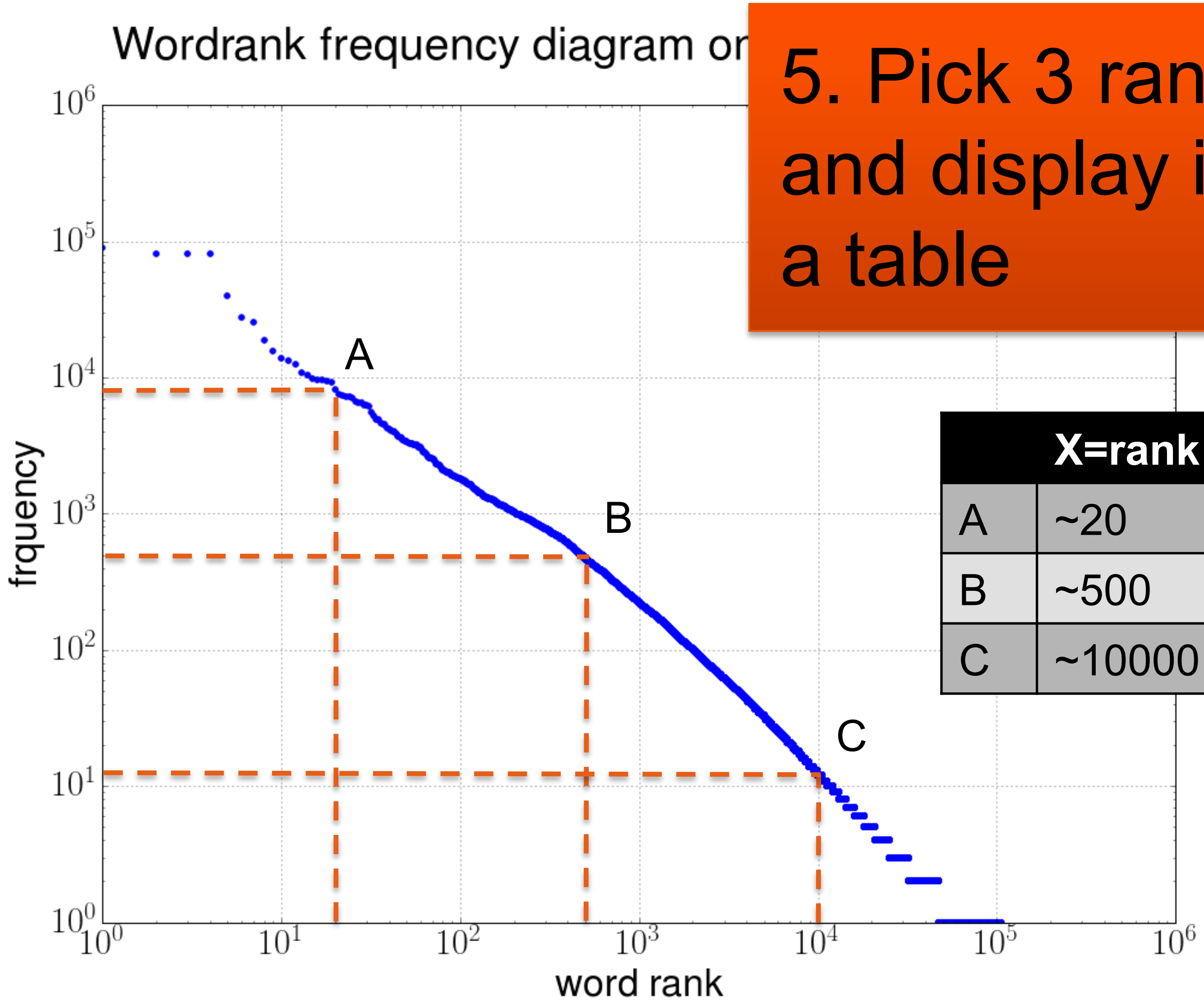
6 Steps of mastering reading (log-log) plots



5. Pick 3 random points and display its values in a table

	X=rank	Y=frequency
A	20	~8000
B	500	~500

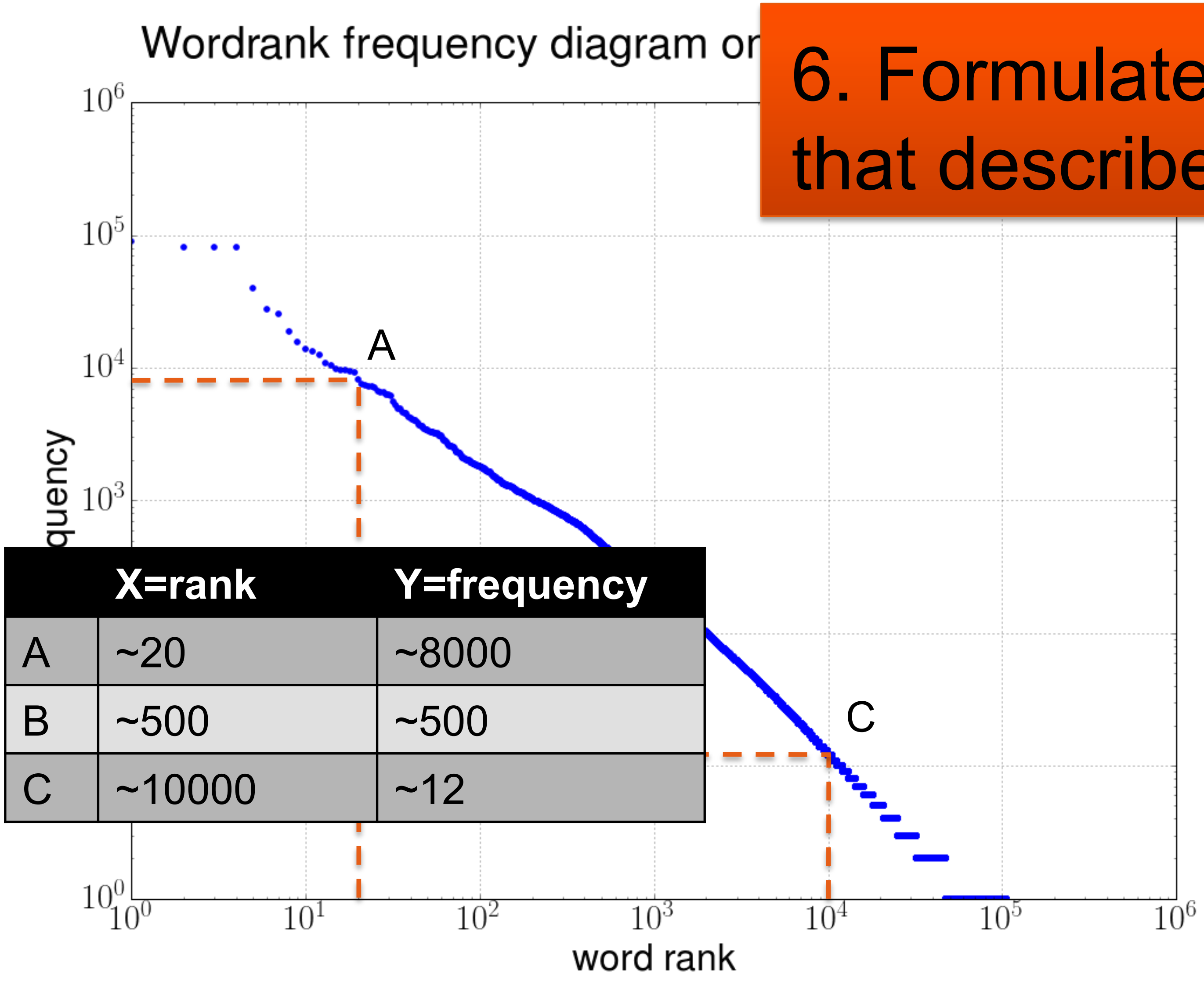
6 Steps of mastering reading (log-log) plots



5. Pick 3 random points and display its values in a table

	X=rank	Y=frequency
A	~20	~8000
B	~500	~500
C	~10000	~12

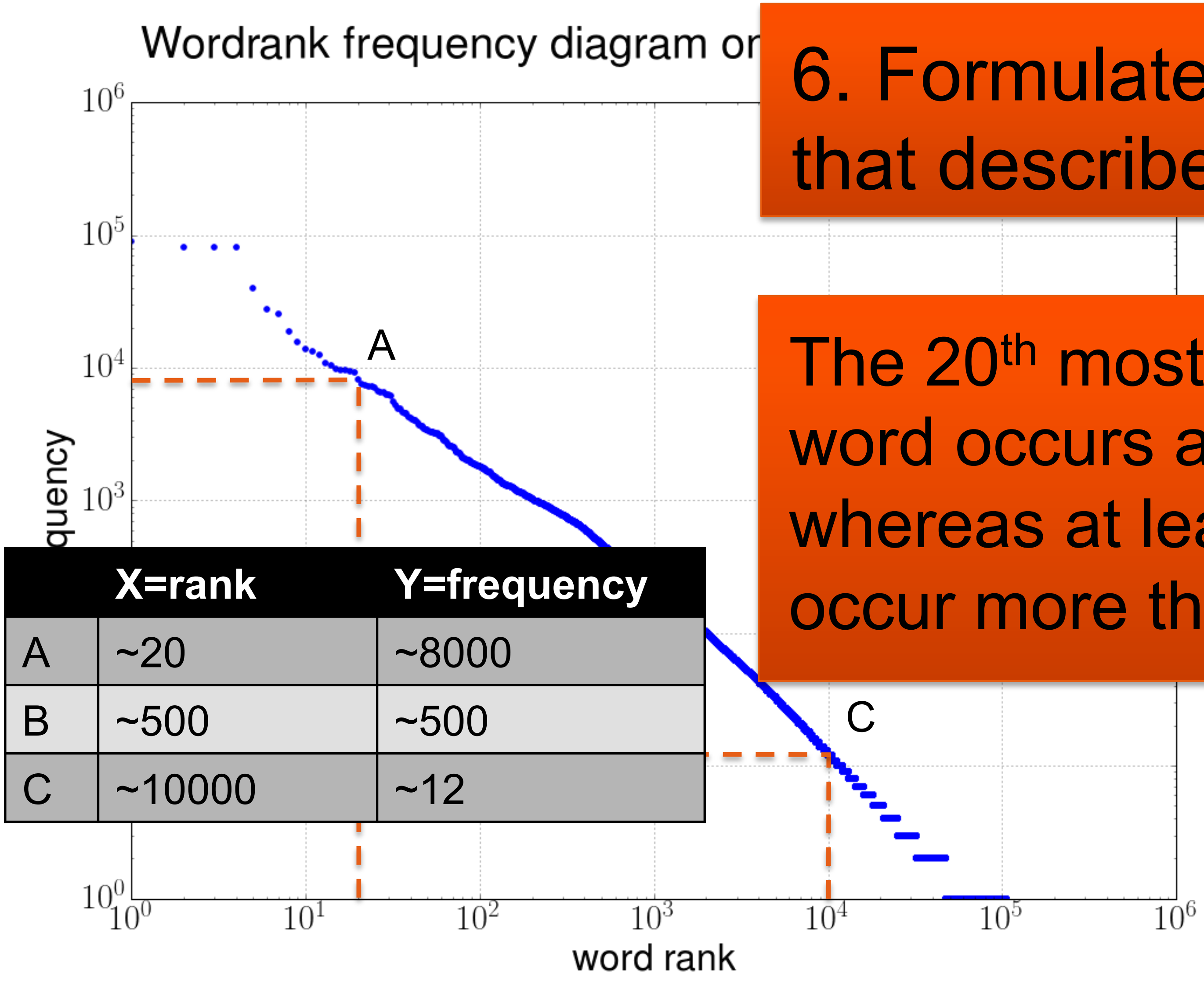
6 Steps of mastering reading (log-log) plots



6. Formulate a sentence that describes the plot

	X=rank	Y=frequency
A	~20	~8000
B	~500	~500
C	~10000	~12

6 Steps of mastering reading (log-log) plots



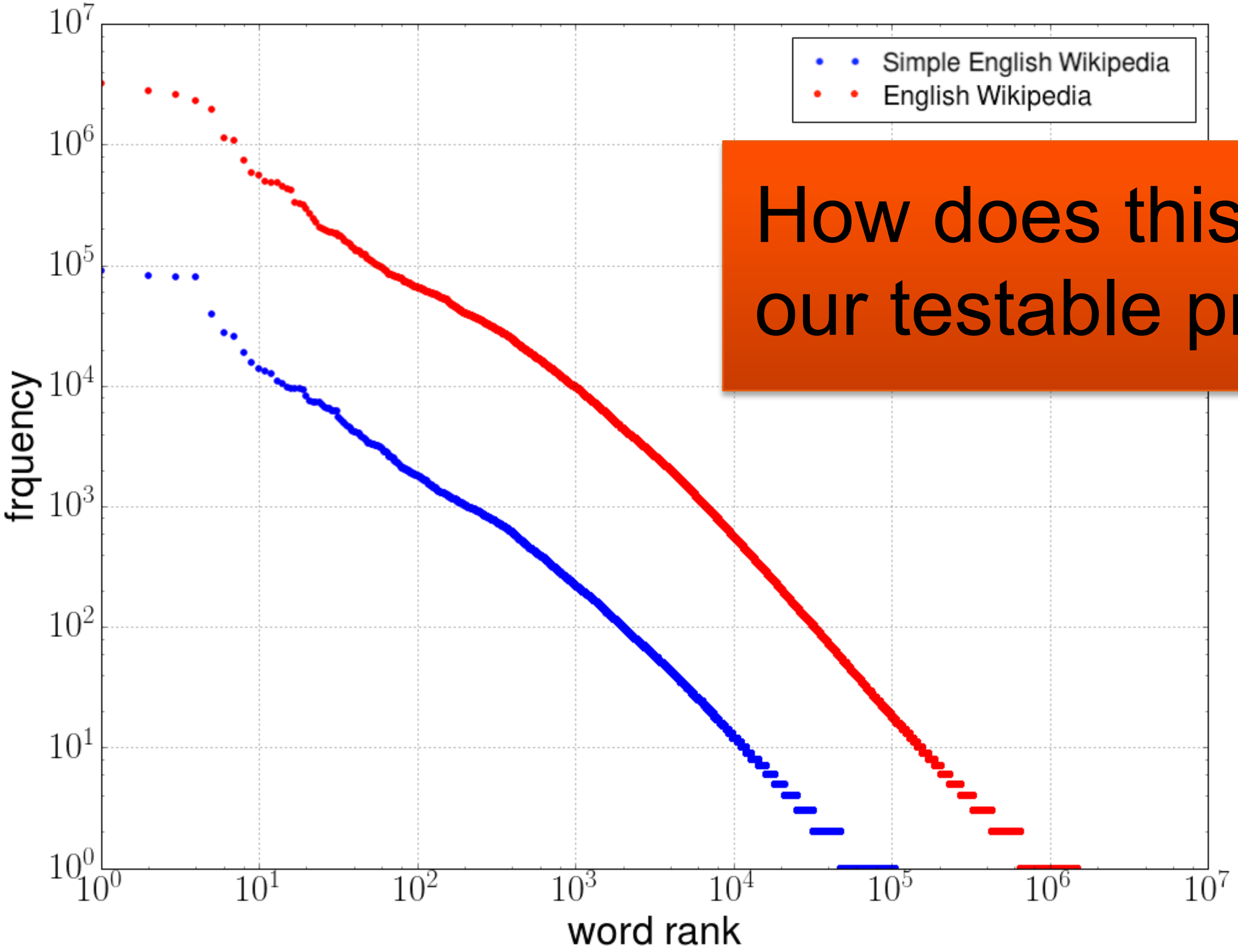
6. Formulate a sentence that describes the plot

	X=rank	Y=frequency
A	~20	~8000
B	~500	~500
C	~10000	~12

The 20th most frequent word occurs about 8k times whereas at least 10k words occur more than ten times.

Visualizing both data sets

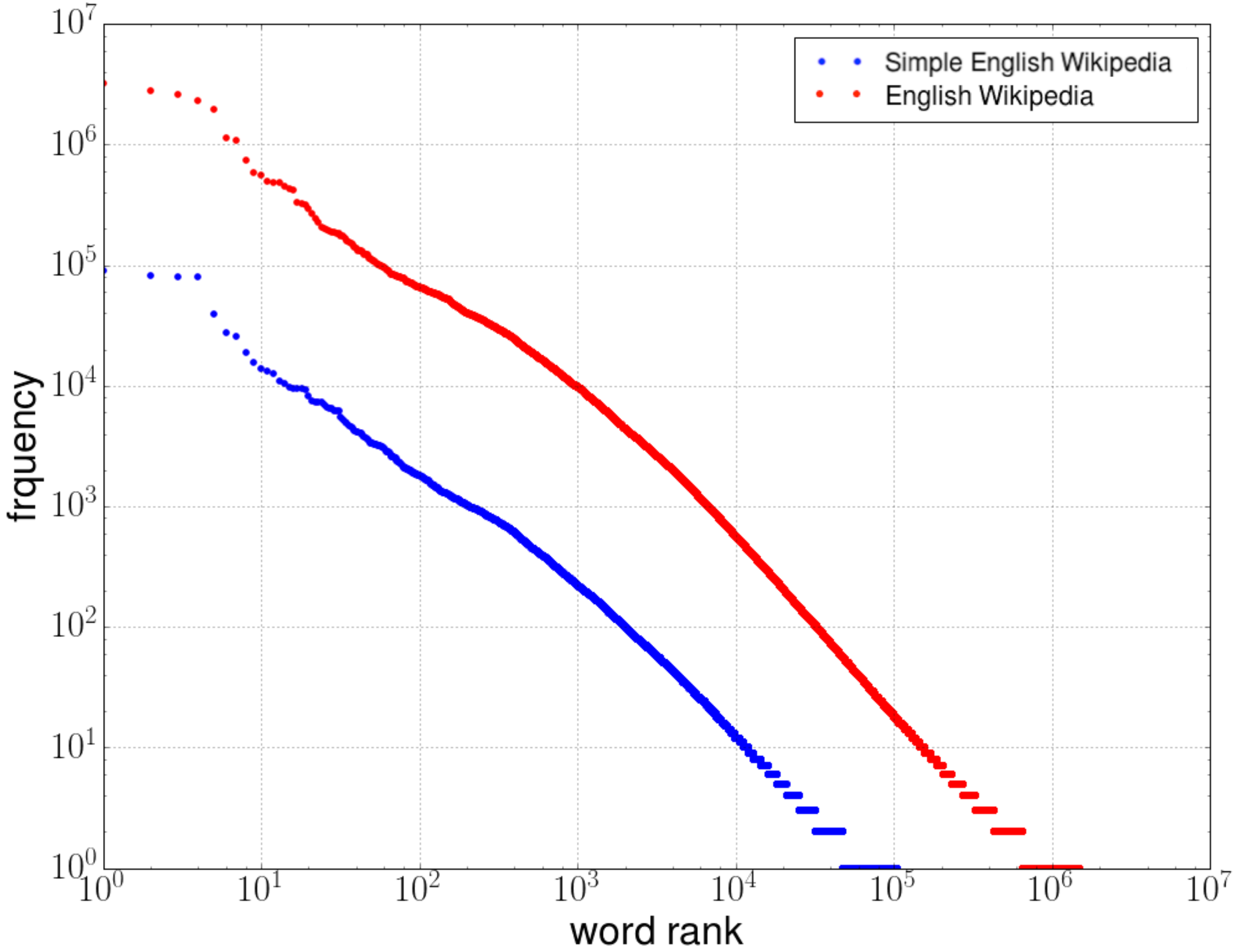
Wordrank frequency diagram on Wikipedia data sets (log-log scale)



How does this support our testable prediction?

Beware word order not the same!

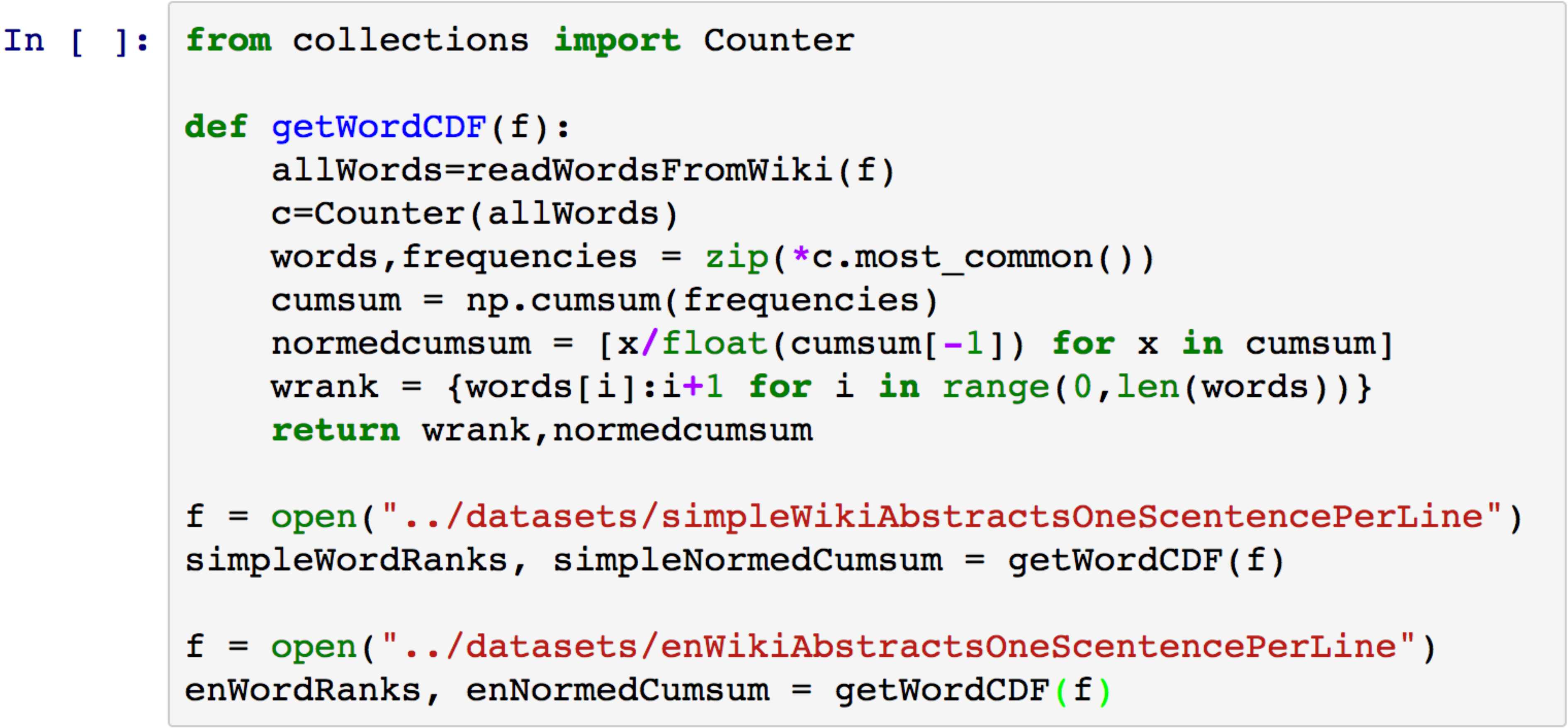
Wordrank frequency diagram on Wikipedia data sets (log-log scale)



Comparing the top 10 words

	Simple English Wiki	English Wiki
1st	the	the
2nd	is	of
3rd	a	in
4th	of	a
5th	in	is
6th	and	and
7th	it	was
8th	was	to
9th	to	by
10th	an	it
Average frequency	**20.04**	**57.74**
Median frequency	**1**	**1**

Creating the Cumulative Distribution Function



In [ ]:

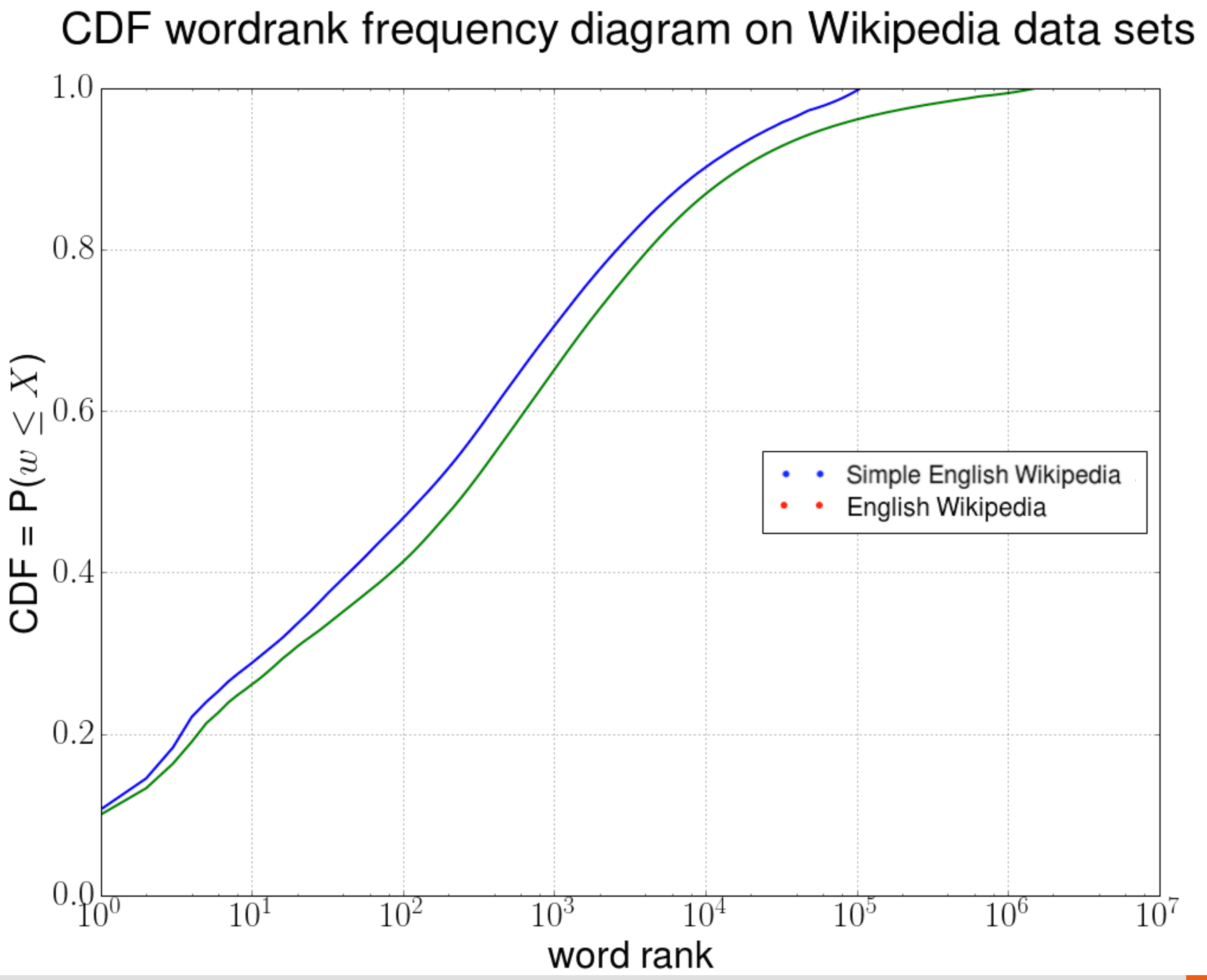
```
from collections import Counter

def getWordCDF(f):
    allWords=readWordsFromWiki(f)
    c=Counter(allWords)
    words,frequencies = zip(*c.most_common())
    cumsum = np.cumsum(frequencies)
    normedcumsum = [x/float(cumsum[-1]) for x in cumsum]
    wrank = {words[i]:i+1 for i in range(0,len(words))}
    return wrank,normedcumsum

f = open("../datasets/simpleWikiAbstractsOneScentencePerLine")
simpleWordRanks, simpleNormedCumsum = getWordCDF(f)

f = open("../datasets/enWikiAbstractsOneScentencePerLine")
enWordRanks, enNormedCumsum = getWordCDF(f)
```

Visualizing the CDF!



Now lets be critical!

- Understanding 80% of all words does not necessarily mean that one understands 80% of the text

- Or do you understand the meaning of:
  - But it is her Schadenfreude

- English Wikipedia Corpus much bigger / more articles than Simple English
  - Comparing apples and peaches?

- Counting is ambiguous: Various forms for the “same” word like:
  - word, words
  - be, was, were, am, is
  - have, has, had

We could change the question a little bit

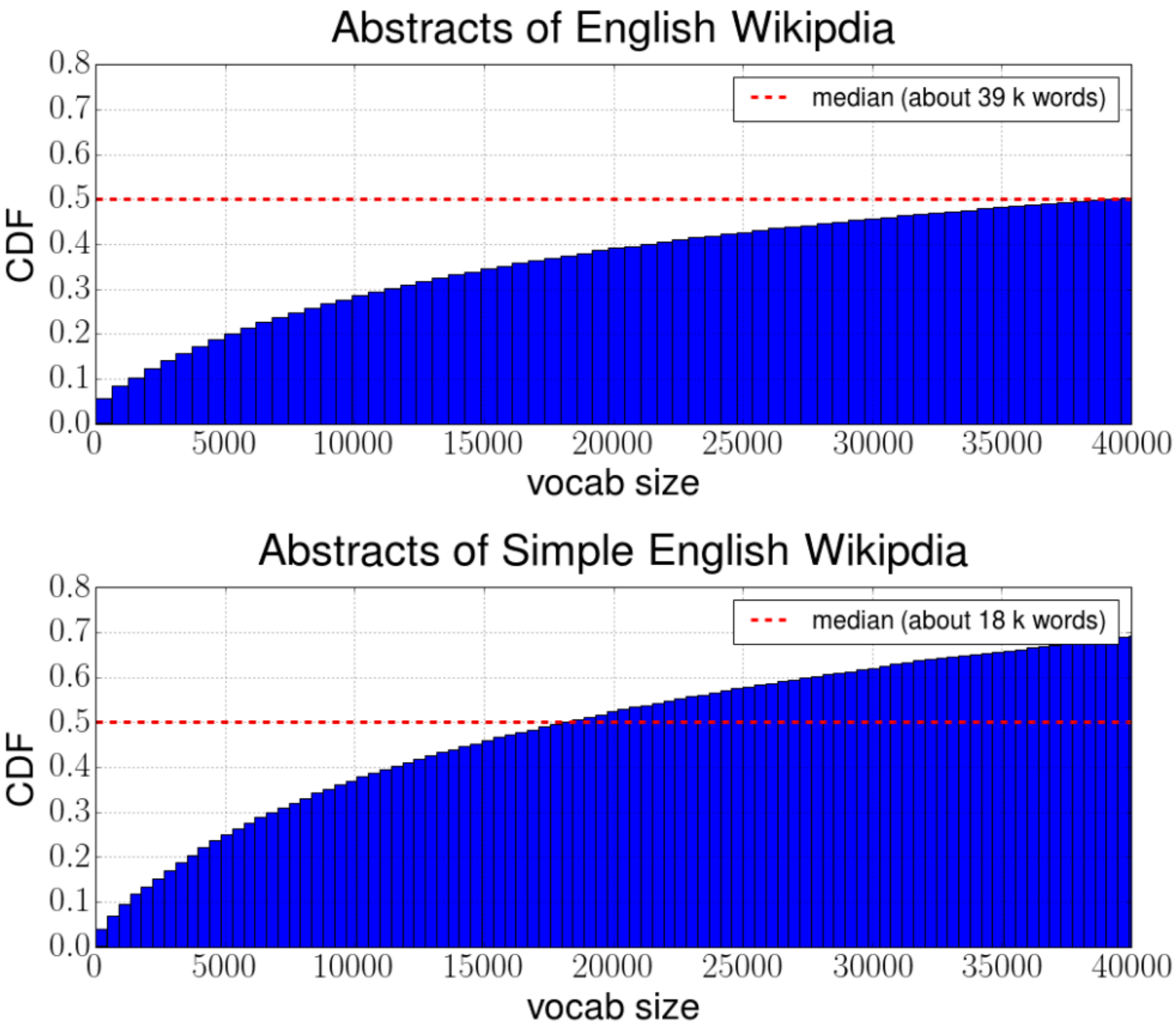
- How many words does one need to know all words in a given sentence?

- Can be done with the same tools and techniques

- Lets dig directly into the results

Repeat on sentences instead of words

CDF for understanding all words in a scentince given a vocab of top popular words of a certain size



Thank you for your attention!

Contact:
Rene Pickhardt
Institute for Web Science and Technologies
Universität Koblenz-Landau
rpickhardt@uni-koblenz.de

WeST
People and Knowledge Networks

Copyright:

- This Slide deck is licensed under creative commons 3.0. share alike attribution license. It was created by Rene Pickhardt. You can use share and modify this slide deck as long as you attribute the author and keep the same license. All graphics have been self made (unless otherwise stated)