LG 467 Computers in Linguistics

[1-2021] Topic 4: Corpus Exploration

Sakol Suethanapornkul

Corpus

Corpus = a 'body' of written text or transcribed speech

Aims:
- to represent a domain of language use
- to allow for an analysis of actual patterns of use

Features:
- usually but not necessarily purposefully collected
- usually but not necessarily structured
- usually but not necessarily annotated

Issues in corpus design

Design:	Representativeness & balance
Collection:	Access, accuracy, & adequacy
Consideration:	Copyright, permission, & consent

Documentation:

- how corpus is compiled; what metadata are collected

Annotation:

- what linguistic analysis is done on the text

Some examples…

A sample of *representative* general classes of corpora

1. Small, 1-5 million-word 1st-gen corpora like the ***Brown Corpus***
2. Moderately sized, 2nd-gen, genre-balanced corpora such as the 100-million-word ***BNC***
3. Larger, more up-to-date (but still genre-balanced) corpora, such as 1-billion-word ***COCA***
4. Extremely large text archives, such as ***Google Books***, and so on

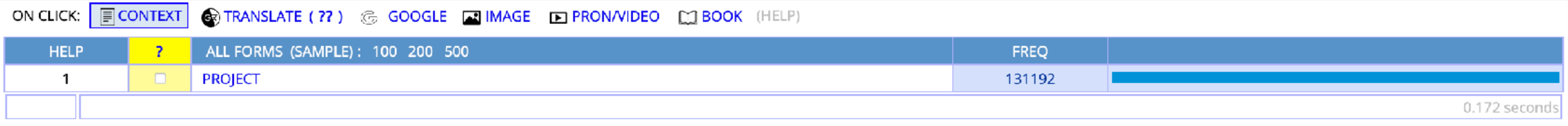
But seriously, why?

Corpus provides useful data on various linguistic phenomena:

Areas	**Examples**
Lexical	Frequency and distribution of specific words and phrases Lists of all common words in a language or genre
Morphology	Processes involving word formation (e.g., nouns formed with suffixes *ism) Contrasts in the use of grammatical alternative (e.g., HAVE + proven/proved)
Grammar/syntax	High-frequency grammatical features, like modals, passives, etc. Less frequent grammatical variation, such as choices with verb subcategorization
Semantics	Collocates (generally) as a guide to meaning and usage Semantic prosody (e.g., the types of words preceding the verg budge)

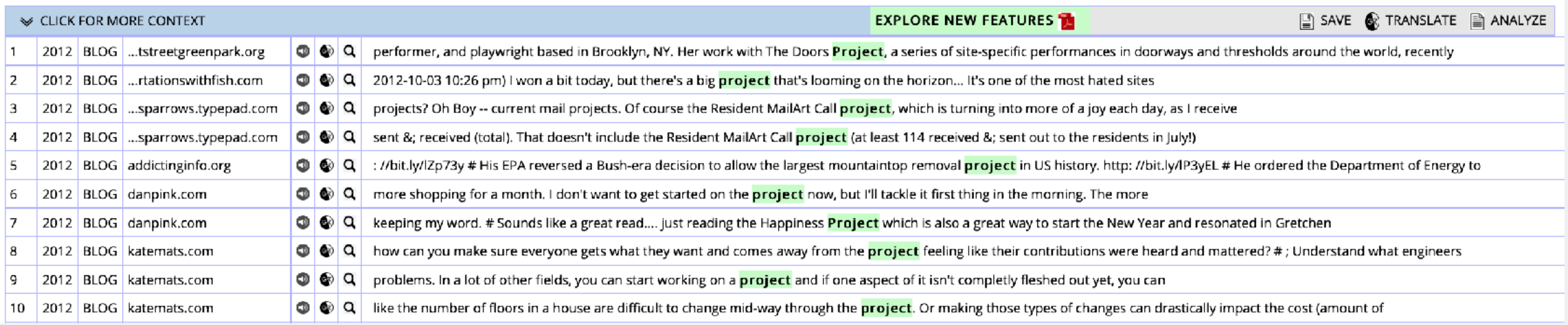
But seriously, why?

But all of these phenomena must be deduced from frequencies



HELP	?	ALL FORMS (SAMPLE) : 100 200 500	FREQ	
1	□	PROJECT	131192	

0.172 seconds



1	2012	BLOG	...tstreetgreenpark.org	performer, and playwright based in Brooklyn, NY. Her work with The Doors **Project**, a series of site-specific performances in doorways and thresholds around the world, recently
2	2012	BLOG	...rtationswithfish.com	2012-10-03 10:26 pm) I won a bit today, but there's a big **project** that's looming on the horizon... It's one of the most hated sites
3	2012	BLOG	...sparrows.typepad.com	projects? Oh Boy -- current mail projects. Of course the Resident MailArt Call **project**, which is turning into more of a joy each day, as I receive
4	2012	BLOG	...sparrows.typepad.com	sent &; received (total). That doesn't include the Resident MailArt Call **project** (at least 114 received &; sent out to the residents in July!)
5	2012	BLOG	addictinginfo.org	: //bit.ly/lZp73y # His EPA reversed a Bush-era decision to allow the largest mountaintop removal **project** in US history. http: //bit.ly/lP3yEL # He ordered the Department of Energy to
6	2012	BLOG	danpink.com	more shopping for a month. I don't want to get started on the **project** now, but I'll tackle it first thing in the morning. The more
7	2012	BLOG	danpink.com	keeping my word. # Sounds like a great read.... just reading the Happiness **Project** which is also a great way to start the New Year and resonated in Gretchen
8	2012	BLOG	katemats.com	how can you make sure everyone gets what they want and comes away from the **project** feeling like their contributions were heard and mattered? # ; Understand what engineers
9	2012	BLOG	katemats.com	problems. In a lot of other fields, you can start working on a **project** and if one aspect of it isn't completly fleshed out yet, you can
10	2012	BLOG	katemats.com	like the number of floors in a house are difficult to change mid-way through the **project**. Or making those types of changes can drastically impact the cost (amount of

Frequencies

Speaking of frequencies…

- **Token** means individual occurrence of a word
- **Type** means instance of a unique word form

The man saw the girl with the telescope

Type may refer to lexeme or individual word form

- *run, runs, ran, running*: 1 or 4 types?

Frequencies

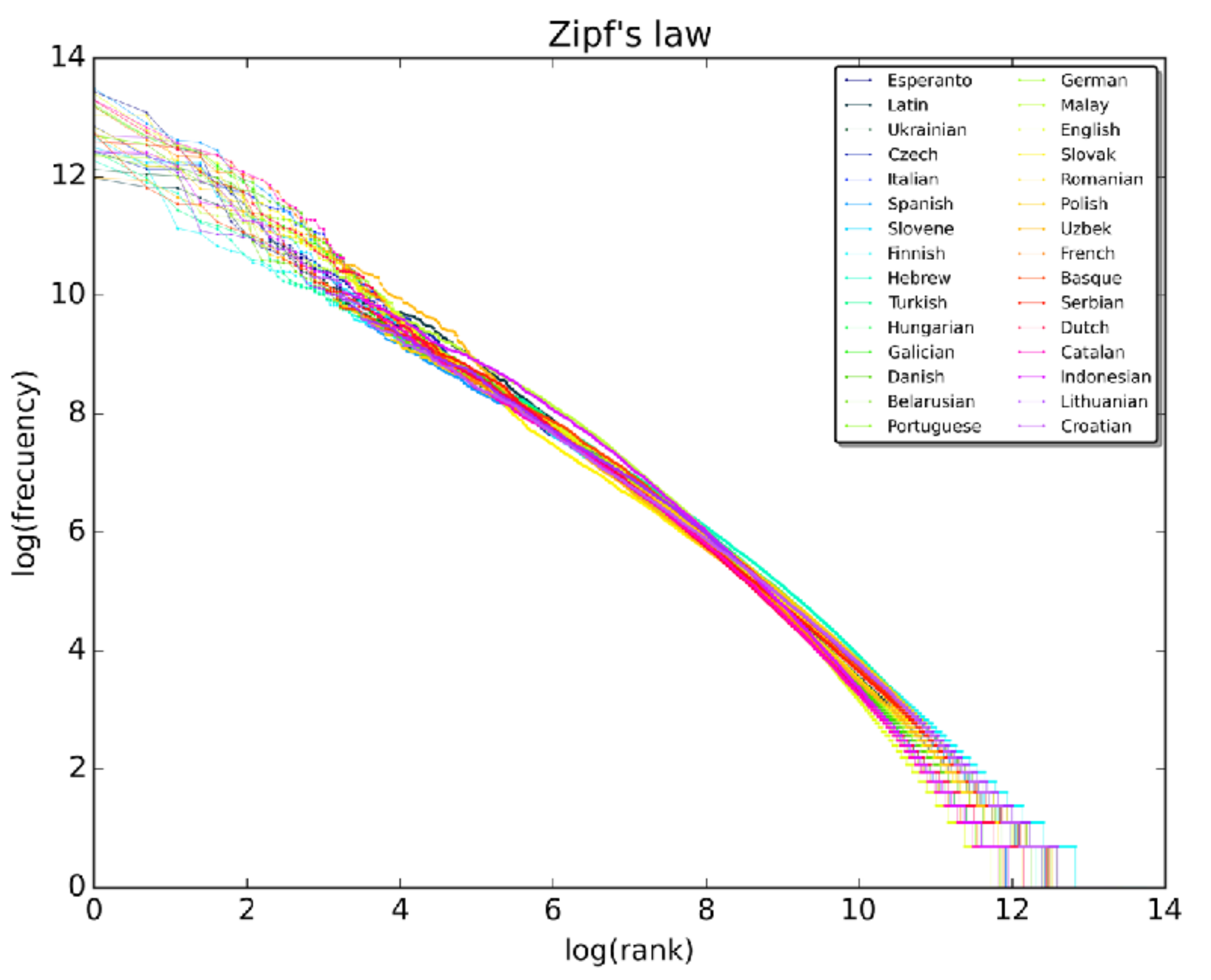
Speaking of frequencies…

- **Type/token ration (TTR)**
    - Number of types ÷ number of tokens
    - This indicates lexical variation in text
- **Hapax legomenon** (pl: hapax legomena, hapax & hapaxes)
    - Tokens that occur only once (in text, etc.)

Zipf’s Law

Language obeys **Zipf’s Law**. That is, a word's frequency is inversely proportional to its rank in the frequency table.

$$f(r) \propto \frac{1}{r^a}; a \approx 1$$



Source: Wikipedia

Corpus Exploration

Corpora in NLTK

NLTK comes bundled with many corpora. Let's focus on one

Brown Corpus (the 1st 1-million word corpus)

- complied by Francis and Kučera at Brown University
- consisting of American English texts printed 1961
- considered to be first general corpus with diverse genres (500 texts, 2000 words each)

Corpora in NLTK

Accessing the corpus in NLTK is extremely easy:

Code 5.3

```
from nltk.corpus import brown

brown.fileids()         #files of corpus
brown.categories()      #categories of corpus
brown.raw()             #raw content of corpus
brown.raw(fileids = []) #raw content of specified files
brown.raw(categories = [])
brown.words()           # word of the whole corpus
brown.words(fileids = [])
brown.words(categories = [])
brown.sents()           #sentences of the whole corpus
```

Corpora in NLTK

Let's use whatever we have learned to deal with raw files!

Code 5.4

```
from nltk.tokenize import word_tokenize

textfile = brown.raw(fileids = 'ca01')
tokens = word_tokenize(textfile)
print(tokens[0:21])

tok = []
for item in tokens:
    raw = re.search(r"([^ ]+)(?=\/)", item)
    if raw:
        tok.append(raw.group())
# We need positive lookahead to match whatever before /
# But there are some tokens with no POS tag, match = None
```

List comprehension

You can "filter" items in a list with list comprehension

Code 5.5

```
# Let's say we want words whose length > 5
long_words = []
for w in tok:
    if len(w) > 5:
        long_words.append(w)


# [word for word in list if .....]
long = [w for w in tok if len(w) > 5]
```

For every "word" in tok

when its length is greater than 5

List comprehension

You can combine multiple conditions with and or or

Code 5.6

```
[w for w in tok if len(w) > 8 and w.endswith('es')]
[w for w in tok if len(w) > 8 or w.endswith('es')]
```

Quiz: Filter out nouns with “-tion" ending & length > 8

Filter out words that starts with vowels & whose length > 3

List comprehension

You can filter and transform the list at the same time

Code 5.7

```
# [f(x) for x in list if.....]

[w.lower() for w in tok if len(w) > 5]
[w+"/NN" for w in tok if w.endswith('tion')]
```

Corpus exploration

Now, let's get back to the Brown Corpus. NLTK provides some useful tools for corpus work

Code 5.8

```
from nltk.book import FreqDist

# If you get an error
import nltk
nltk.download(“book")

all_words = FreqDist([t.lower() for t in tok])

all_words.most_common(10)
```

Corpus exploration

But notice, the most frequent word is "the." How can we do better?

```
stop = ['a', 'an', 'the', 'in', 'on', 'at', 'to',
'for', 'of', 'and', ‘.']

no_stop = [t.lower() for t in tok]
no_stop = [t for t in no_stop if t not in stop]

words = FreqDist(no_stop)
words.most_common(10)
```

Code 5.8

[Continued]