

# CS4740 Natural Language Processing

---

- Today
  - Introduction to an important class of statistical methods in NLP: generative models
  - Writing critiques
    - first critique out: today
    - first critique due: Weds night

but first...

# Administrative stuff

---

- Course website:
  - <http://www.cs.cornell.edu/courses/cs4740/2015fa/>
- Piazza
  - Signup link:
    - [piazza.com/cornell/fall2015/cs4740](https://piazza.com/cornell/fall2015/cs4740)
  - Class link:
    - [piazza.com/cornell/fall2015/cs4740/home](https://piazza.com/cornell/fall2015/cs4740/home)

# In the last class, we learned that...

---

- A. NLP is hard (for computers)
- B. NLP is pretty cool!
- C. All of the above
- D. None of the above
- E. Whatever.

# CS4740 Natural Language Processing

---

Next few lectures...Language Modeling

- Today

- Introduction to generative models of language

- » What are they?

- » Why they're important

- » An aside: issues for counting words

- » Another aside: statistics of NL

# What are generative models of NL?

---

- Explicitly model the process of producing/generating language
- Models of **word prediction**

# How would you finish each of these sentence fragments?

---

- Word prediction task
  - *Once upon a...*
  - *I'd like to make a collect...*
  - *Let's go outside and take a...*
- Generative models can assign probabilities to
  - Possible next words
  - Entire sequences of words

# Why are word prediction models important?

---

- Augmentative communication systems
  - For the disabled, to predict the next words the user wants to “speak”
- Computer-aided education
  - System that helps kids learn to read (e.g. Mostow et al. system)
- Speech recognition
- Context-sensitive spelling correction
- Important in real-life situations...
  - Miss words in a conversation, lecture, movie, etc.

# Why are word prediction models important?

---

- Can be used to assign a probability to the next word in an incomplete sentence
- Closely related to the problem of computing the probability of a sequence of words
  - Useful for part-of-speech tagging, probabilistic parsing, ...



---

The need for models of word prediction in NLP has not been uncontroversial

But it must be recognized that the notion “probability of a sentence” is an entirely useless one, under any known interpretation of this term. - Noam Chomsky (1969)

Every time I fire a linguist the recognition rate improves.

- Fred Jelinek (IBM speech group, 1988)

# Word prediction gone amok

---

- Seinfeld Sentence Finisher

<http://www.youtube.com/watch?v=01teZKTYjQA&feature=related>

# Word prediction gone awry

---



Woody Allen's "Take the Money and Run"

<http://www.tcm.com/mediaroom/video/224555/Take-the-Money-and-Run-Movie-Clip-Gub.html>

# N-gram models

---

- Use the previous  $N-1$  words to predict the next word
  - 2-gram: bigram
  - 3-gram: trigram
  - 1-gram: unigram
- In speech recognition, these statistical models of word sequences are referred to as a **language model**

# Want to use n-gram models to...

---


- Determine the next word in a sequence
  - Probability distribution across all words in the language
  - $P(w_n | w_1 w_2 \dots w_{n-1})$
- Determine the probability of a sequence of words
  - $P(w_1 w_2 \dots w_{n-1} w_n)$

---

## Next...Language Modeling

- Introduction to generative models of language

- » What are they?
- » Why they're important

-  » Issues for counting words
- » Statistics of natural language
- » Unsmoothed n-gram models

# Counting words in corpora

---

- Ok, so how many words are in this sentence?
- Depends on whether or not we treat punctuation marks as words
  - Important for many NLP tasks
    - » Grammar-checking, spelling error detection, author identification, part-of-speech tagging
- Spoken language corpora
  - Utterances don't usually have punctuation, but they do have other phenomena that we might or might not want to treat as words
    - » I do uh main- mainly business data processing
  - Fragments
  - Filled pauses
    - » *um* and *uh* behave more like words, so most speech recognition systems treat them as such

# Counting words in corpora

---

- Capitalization

- Should *They* and *they* be treated as the same word?
  - » For most statistical NLP applications, they are
  - » Sometimes capitalization information is maintained as a feature
    - » E.g. spelling error correction, part-of-speech tagging

- Inflected forms

- Should *walks* and *walk* be treated as the same word?
  - » No...for most n-gram based systems



# Counting words in corpora

---

- Need to distinguish the counting of
  - word **types**
    - » the number of distinct words
  - word **tokens**
    - » the number of running words
- Example
  - *All for one and one for all.*
  - 8 tokens (counting punctuation)
  - 6 word types (assuming capitalized and uncapitalized versions of the same token are treated separately)

---

- Introduction to generative models of language

- » What are they?
- » Why they're important
- » Issues for counting words
- » **Statistics of natural language**
- » Unsmoothed n-gram models

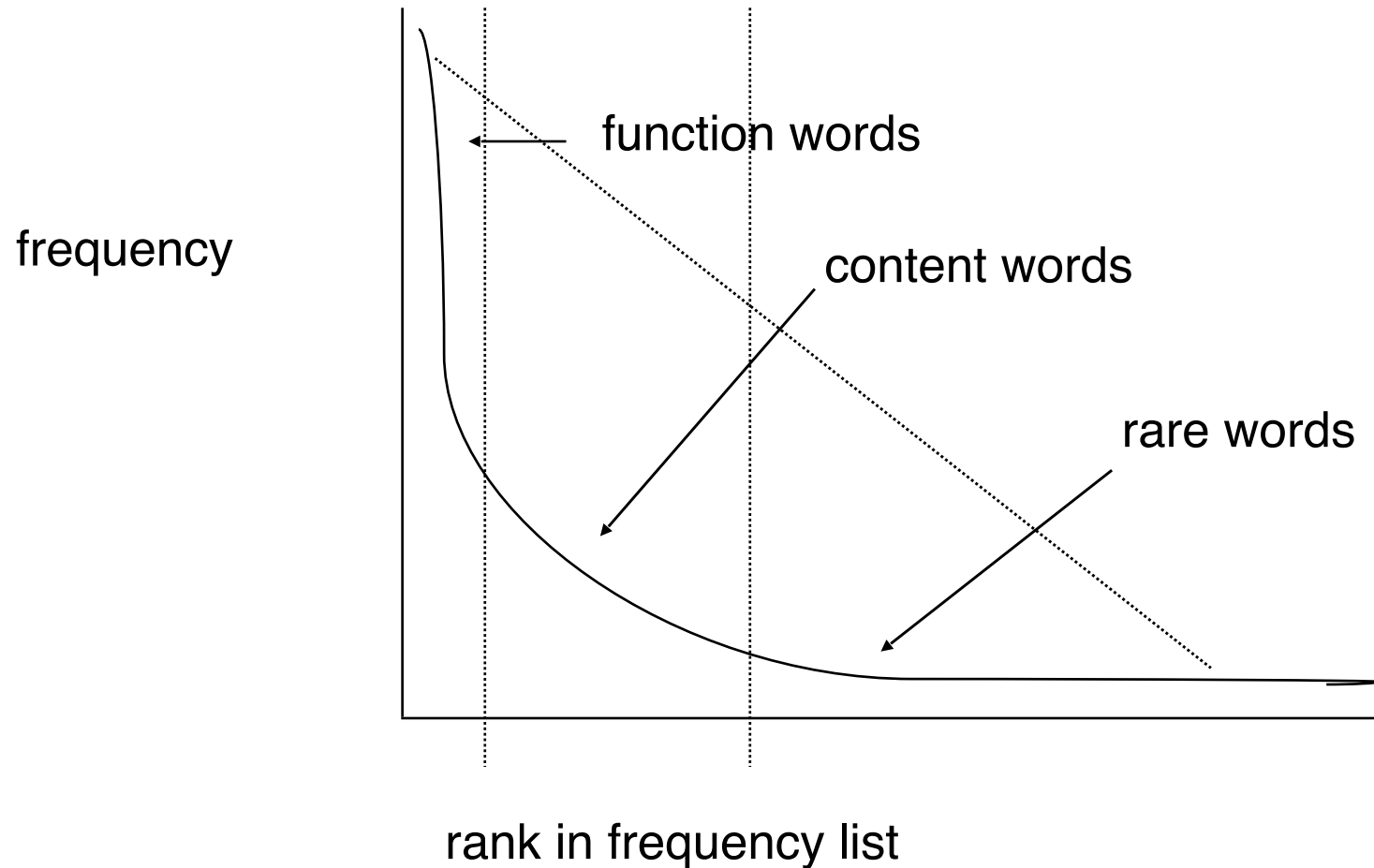
# How many words are there in English?

---

- **Option 1: count the word entries in a dictionary**
  - OED: 600,000
  - American Heritage (3<sup>rd</sup> edition): 200,000
  - Actually counting *lemmas* not *wordforms(word types)*
- **Option 2: estimate from a corpus**
  - Switchboard: 2.4 million word tokens; 20,000 word types
  - Shakespeare's complete works: 884,647 word tokens; 29,066 word types
  - Brown corpus: 1 million word tokens; 61,805 word types; 37,851 lemma types
  - Brown et al. 1992: 583 million word tokens, 293,181 word types

# How are they distributed?

---



# Statistical Properties of Text

---

- Zipf's Law relates a term's frequency to its rank
  - frequency  $\propto 1/\text{rank}$
  - There is a constant  $k$  such that  $\text{freq} * \text{rank} = k$
- The most frequent words in one corpus may be rare words in another corpus
  - Example: “computer” in CACM vs. National Geographic
- Each corpus has a different, fairly small “working vocabulary”

These properties hold in a wide range of languages

# Zipf's Law (*Tom Sawyer*)

---

Word	Freq. ( $f$ )	Rank ( $r$ )	$f \cdot r$	Word	Freq. ( $f$ )	Rank ( $r$ )	$f \cdot r$
the	3332	1	3332	turned	51	200	10200
and	2972	2	5944	you'll	30	300	9000
a	1775	3	5235	name	21	400	8400
he	877	10	8770	comes	16	500	8000
but	410	20	8400	group	13	600	7800
be	294	30	8820	lead	11	700	7700
there	222	40	8880	friends	10	800	8000
one	172	50	8600	begin	9	900	8100
about	158	60	9480	family	8	1000	8000
more	138	70	9660	brushed	4	2000	8000
never	124	80	9920	sins	2	3000	6000
Oh	116	90	10440	Could	2	4000	8000
two	104	100	10400	Applausive	1	8000	8000

# Zipf's Law

---

- Behavior occurs in a surprising variety of situations
  - References to scientific papers
  - Web page in-degrees, out-degrees
  - Royalties to pop-music composers
  - English verb polysemy
- For NLP, Zipf's Law is useful as a rough description of the frequency distribution of words in human languages