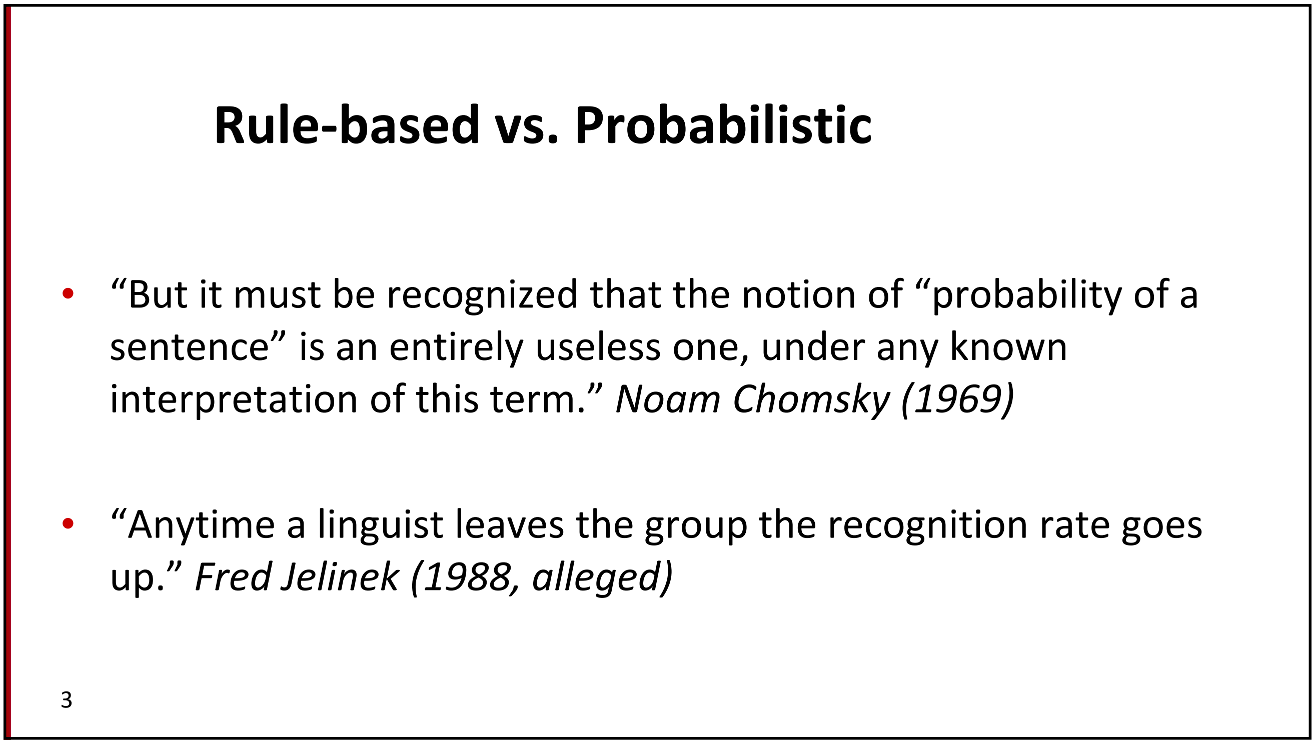
Language Modeling with N-grams

Chapter 3

(3.1-3.4)

Review

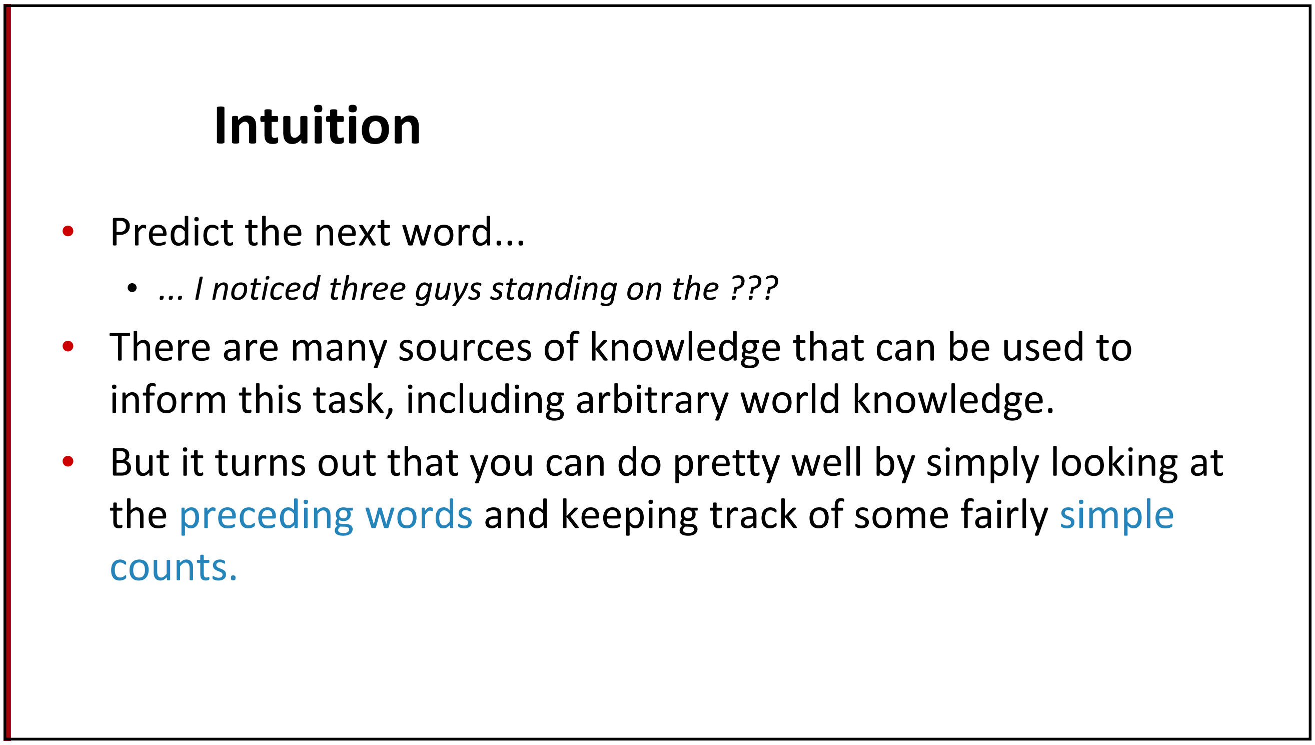
- Text Normalization
  - Why?
  - How computationally?
  - Example tasks?



Rule-based vs. Probabilistic

- “But it must be recognized that the notion of “probability of a sentence” is an entirely useless one, under any known interpretation of this term.” *Noam Chomsky (1969)*
- “Anytime a linguist leaves the group the recognition rate goes up.” *Fred Jelinek (1988, alleged)*

3



Intuition

- Predict the next word...
  - *... I noticed three guys standing on the ???*
- There are many sources of knowledge that can be used to inform this task, including arbitrary world knowledge.
- But it turns out that you can do pretty well by simply looking at the preceding words and keeping track of some fairly simple counts.

Word Prediction

- We can formalize this task using what are called *N*-gram models.
- *N*-grams are token sequences of length *N*.
- This example contains what 2-grams (aka bigrams)?
  - *I notice three guys standing on the*
- Given knowledge of counts of N-grams such as these, we can guess likely next words in a sequence.

N-Gram Models

- More formally, we can use knowledge of the counts of *N*-grams to assess the *conditional probability* of candidate words as the next word in a sequence.
- Or, we can use them to assess the *probability* of an entire sequence of words.
  - Pretty much the same thing as we'll see...

Probability

Quick Review

Different Kinds of Statistics

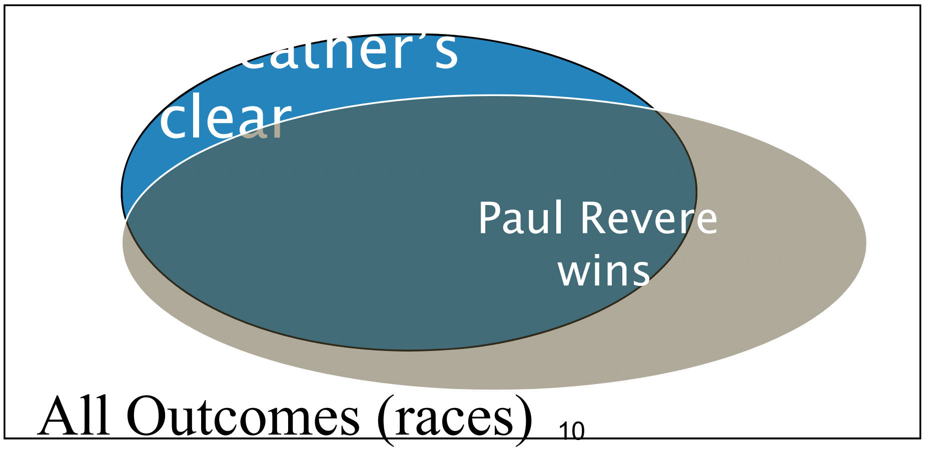
- **descriptive:** mean Pitt QPA (or median)
- **confirmatory:** statistically significant?
- **predictive:** wanna bet?
  - N-grams

Notation

p(Paul Revere wins | weather’s clear) = 0.9

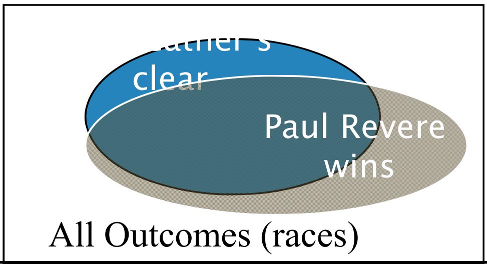
p is a function on sets of “outcomes”

p(win | clear) ≡ p(win, clear) / p(clear)



p is a function on sets of “outcomes”

$$\underbrace{p(\text{win} \mid \text{clear})}_{\text{syntactic sugar}} \equiv p(\underbrace{\text{win, clear}}_{\substack{\text{logical conjunction} \\ \text{of predicates}}}) \;/\; p(\underbrace{\text{clear}}_{\substack{\text{predicate selecting} \\ \text{races where} \\ \text{weather's clear}}})$$



p measures total probability of a set of outcomes

Required Properties of p (axioms)

most of the

- $p(\varnothing) = 0$ $\qquad$ $p(\text{all outcomes}) = 1$
- $p(X) \le p(Y)$ for any $X \subseteq Y$
- $p(X) + p(Y) = p(X \cup Y)$ provided $X \cap Y = \varnothing$

e.g., p(win & clear) + p(win & ¬clear) = p(win)

Commas denote conjunction

p(Paul Revere wins | weather's clear, ground is dry, jockey getting over sprain, Epitaph also in race, Epitaph was recently bought by Gonzalez, race is on May 17, …)

Simplifying Right Side: Backing Off

p(Paul Revere wins | weather's clear, ground ~~is dry, jockey getting over sprain~~, Epitaph also in race, ~~Epitaph was~~ recently ~~bought by Gonzalez, race is on May 17, …~~)

- not exactly what we want but at least we can get a reasonable estimate of it!
- try to *keep* the conditions that we suspect will have the most influence on whether Paul Revere wins

Language Modeling

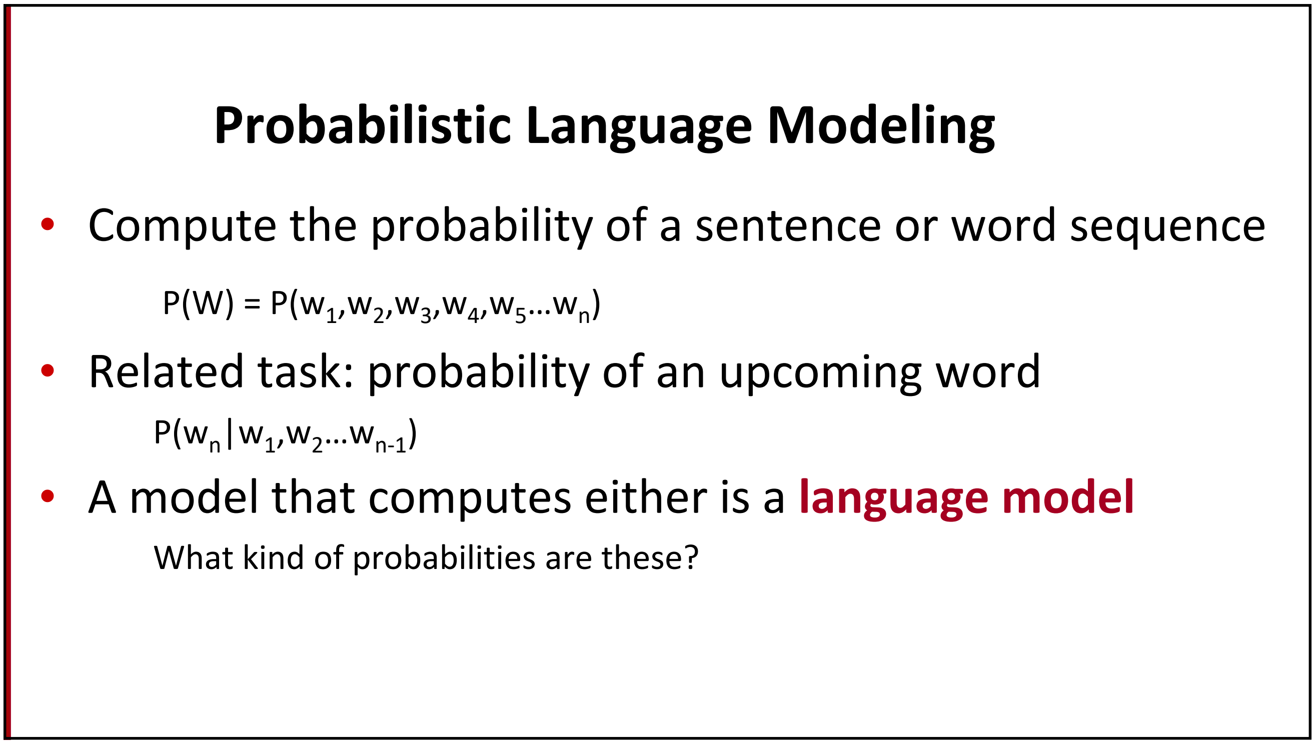
Introduction to N-grams

Probabilistic Language Models

- Goal: assign a probability to a sentence

Why?

- Machine Translation:
  - P(**high** winds tonite) > P(**large** winds tonite)
- Spell Correction
  - The office is about fifteen **minuets** from my house
    - P(about fifteen **minutes** from) > P(about fifteen **minuets** from)
- Speech Recognition
  - P(I saw a van) >> P(eyes awe of an)
- + many more applications



Probabilistic Language Modeling

- Compute the probability of a sentence or word sequence

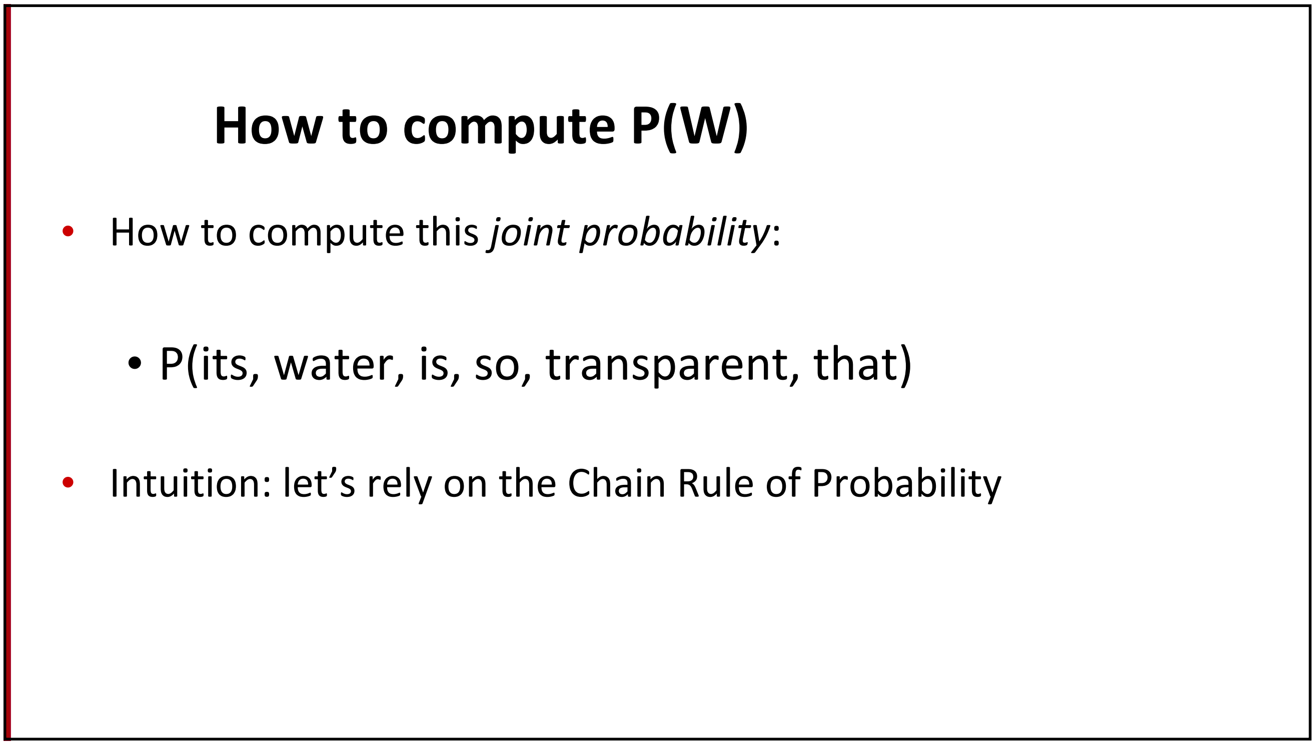
  $P(W) = P(w_1,w_2,w_3,w_4,w_5…w_n)$

- Related task: probability of an upcoming word

  $P(w_n|w_1,w_2…w_{n-1})$

- A model that computes either is a **language model**

  What kind of probabilities are these?



How to compute P(W)

- How to compute this *joint probability*:

  - P(its, water, is, so, transparent, that)

- Intuition: let’s rely on the Chain Rule of Probability

Reminder: The Chain Rule

- Recall the definition of conditional probabilities

  p(B|A) = P(A,B)/P(A) Rewriting: **P(A,B) = P(A)P(B|A)**

  - Independent **p(B|A) = P(B)**

- More variables:

  P(A,B,C,D) = P(A)P(B|A)P(C|A,B)P(D|A,B,C)

- The Chain Rule in General

$$P(x_1,x_2,x_3,\ldots,x_n) = P(x_1)P(x_2|x_1)P(x_3|x_1,x_2)\ldots P(x_n|x_1,\ldots,x_{n-1})$$

The Chain Rule applied to compute joint probability of words in sentence

$$P(w_1 w_2 \ldots w_n) = \prod_i P(w_i \mid w_1 w_2 \ldots w_{i-1})$$

P(“its water is so transparent”) =

P(its) × P(water|its) × P(is|its water)

× P(so|its water is) × P(transparent|its water is so)

How to estimate these probabilities

- Could we just count and divide?

$$P(\text{the} \mid \text{its water is so transparent that}) = \frac{Count(\text{its water is so transparent that the})}{Count(\text{its water is so transparent that})}$$

- No! Too many possible sentences!
- We'll never see enough data for estimating these

Markov Assumption

Andrei Markov

- Simplifying assumption:

$$P(\text{the} \mid \text{its water is so transparent that}) \approx P(\text{the} \mid \text{that})$$

- Or maybe

$$P(\text{the} \mid \text{its water is so transparent that}) \approx P(\text{the} \mid \text{transparent that})$$

Markov Assumption

$$P(w_1 w_2 \square \; w_n) \approx \prod_i P(w_i \mid w_{i-k} \square \; w_{i-1})$$

- In other words, we approximate each component in the product

$$P(w_i \mid w_1 w_2 \square \; w_{i-1}) \approx P(w_i \mid w_{i-k} \square \; w_{i-1})$$

$$P(w_i \mid w_1 w_2 \square \; w_{i-1}) \approx P(w_i \mid w_{i-k} \square \; w_{i-1})$$

- Bigram model (k=1, e.g., context of one so model two words)

Simplest case: Unigram model

$$P(w_1 w_2 \ldots w_n) \approx \prod_i P(w_i)$$

Some automatically generated sentences from a unigram model

```
fifth, an, of, futures, the, an, incorporated, a,
a, the, inflation, most, dollars, quarter, in, is,
mass

thrift, did, eighty, said, hard, 'm, july, bullish

that, or, limited, the
```

Bigram model

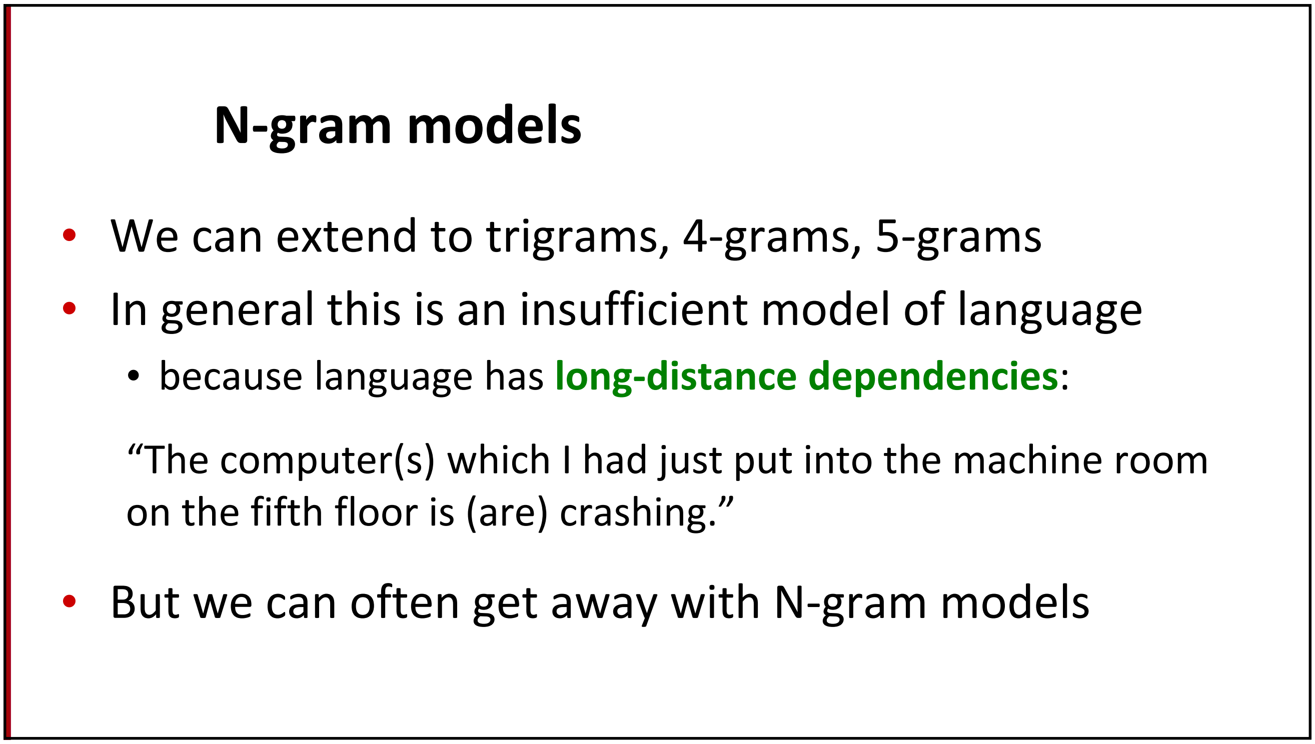
- Condition on the previous word:

$$P(w_i \mid w_1 w_2 \ldots w_{i-1}) \approx P(w_i \mid w_{i-1})$$

```
texaco, rose, one, in, this, issue, is, pursuing, growth, in,
a, boiler, house, said, mr., gurria, mexico, 's, motion,
control, proposal, without, permission, from, five, hundred,
fifty, five, yen

outside, new, car, parking, lot, of, the, agreement, reached

this, would, be, a, record, november
```



N-gram models

- We can extend to trigrams, 4-grams, 5-grams
- In general this is an insufficient model of language
  - because language has **long-distance dependencies**:

    “The computer(s) which I had just put into the machine room on the fifth floor is (are) crashing.”
- But we can often get away with N-gram models