STATISTICAL LEARNING (AND LANGUAGE MODELS)

Lara J. Martin (she/they)
TA: Aydin Ayanzadeh (he)

11/09/2023
CMSC 671

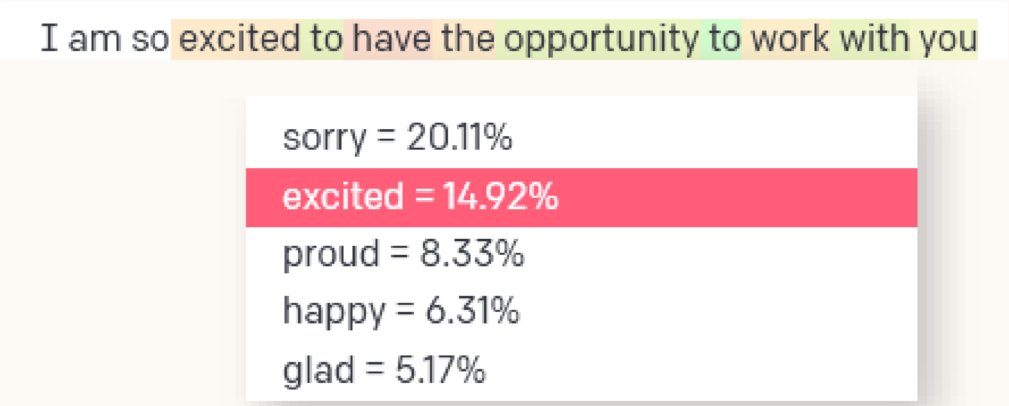
By the end of class today, you will be able to:
- Calculate inference by enumeration & the product rule
- Identify the usefulness of the Markov assumption and the Maximum Likelihood Estimation
- Identify issues with n-grams and MLE

Modified from slides by Dr. Chris Callison-Burch & Dr. Cynthia Matuszek

COURSE SCHEDULE (REMINDER)

- HW 3 is due 11/14
- All project milestones are released (see website)
  - First milestone due 11/16
- Module 4 Presentations are 11/16
  - Summaries due 11/15

RECAP



- Random variable: Unobserved RVs **refer** to a distribution
- Distribution: Exhaustive list of all possible values with their likelihoods
- Joint distribution: The likelihood of two values simultaneously
- Conditional distribution: The likelihood of one RV given another

PROBABILISTIC INFERENCE

Computing the desired probability **from other known probabilities**

- Generally using conditional probabilities
  - P(on time | no reported accidents) = 0.90
  - The agent's *beliefs* given evidence

- Probabilities change with **new evidence**
  - P(on time | no accidents, 5AM) = 0.95
  - P(on time | no accidents, 5AM, raining) = 0.80
  - New evidence → update beliefs

INFERENCE BY ENUMERATION

1. Find the relevant datapoints consistent with the evidence

   E.g., when it was raining and I was on time

2. Sum across all the *h*'s to get the joint probability of the query and the evidence

   E.g., total of *all* the times I was on time when it was raining

3. Normalize i.e., divide each instance by the sum of them all

   E.g., divide by the total across all queries (on time, not on time) with the same evidence (raining, etc.)

$$P(Q, e_1 \dots e_k) = \sum_{h_1 \dots h_r} P(Q, h_1 \dots h_r, e_1 \dots e_k)$$

$$P(Q|e_1 \dots e_k) = \frac{P(Q, e_1 \dots e_k)}{\sum_q P(Q, e_1 \dots e_k)}$$

With:

- "evidence" variables $E_1 \dots E_k = e_1 \dots e_k$
- "query" variable Q
- "hidden" variables $H_1 \dots H_r$

We want $P(Q|\, e_1 \dots e_k)$

Q in this example is "will I be on time?"

INFERENCE BY ENUMERATION EXAMPLE: LANGUAGE MODEL

$P(was|\ it^*)$

1. Find all of the "it was"'s
2. Sum them up 10
3. Normalize $\frac{10}{12} \approx 0.83$

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, it was the epoch of belief, it was the epoch of incredulity, it was the season of Light, it was the season of Darkness, it was the spring of hope, it was the winter of despair, we had everything before us, we had nothing before us, we were all going direct to Heaven, we were all going direct the other way--in short, the period was so far like the present period that some of its noisiest authorities insisted on its being received, for good or for evil, in the superlative degree of comparison only.

…

(From the beginning of *A Tale of Two Cities*)

HELPFUL RULES: PRODUCT RULE

$$P(y)P(x|y) = P(x,y)$$

$P(W)$

W	P
Rain	0.3
Sun	0.6
Fog	0.1

$P(U|W)$

U	W	P
Umbrella	Rain	0.8
No Umbrella	Rain	0.2
Umbrella	Sun	0.1
No Umbrella	Sun	0.9
Umbrella	Fog	0.3
No Umbrella	Fog	0.7

$\Leftrightarrow$

$P(U,W)$

U	W	P
Umbrella	Rain	0.24
No Umbrella	Rain	0.06
Umbrella	Sun	0.06
No Umbrella	Sun	0.54
Umbrella	Fog	0.03
No Umbrella	Fog	0.07

HELPFUL RULES: CHAIN RULE

- You can extend the product rule for each element of a joint distribution
- It becomes a product of conditional distributions:

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

HELPFUL RULES: BAYES' RULE

Product rule tells us we can turn a joint distribution into:

$$P(x,y) = P(y)P(x|y) = P(x)P(y|x)$$

But Thomas Bayes (1763) figured out we can find the conditional distribution if we know the other parts by using division:

$$P(x|y) = \frac{P(y|x)}{P(y)}P(x)$$

Why is this helpful?

NAÏVE BAYES ALGORITHM

- Estimate the probability of each class:
  - Compute the posterior probability (Bayes rule)

$$\boxed{P(c_i \mid D) = \frac{P(c_i)P(D \mid c_i)}{P(D)}}$$

  - Choose the class with the highest probability
- Assumption of attribute independency (Naïve assumption): Naïve Bayes assumes that all of the attributes are independent.

BACK TO LANGUAGE MODELING

- A **probabilistic language model** computes the probability of a word given a sequence of words (or history). E.g.,:

$$P(w_4 | w_1, w_2, w_3)$$
$$P(best | it, was, the)$$

- We can also calculate the probability of an entire sentence. E.g., for a sentence with n words:

$$P(W) = P(w_1, w_2, w_3, \ldots w_n)$$

WHAT ARE LMS USED FOR?

- Machine translation
- Text generation (summarization, dialog systems, question-answering)
- Spelling correction
- Speech recognition

CALCULATING THE JOINT PROBABILITY IN SENTENCES

$$P(w_1, w_2, w_3, \dots w_n) = \prod_i P(w_i | w_1, w_2, \dots w_{i-1})$$

$$P(it\ was\ the\ best\ of\ times) =$$

$$P(it)\ x$$

$$P(was|it)\ x$$

$$P(the|it\ was)\ x$$

$$P(best|it\ was\ the)\ x$$

$$P(of|it\ was\ the\ best)\ x$$

$$P(times|it\ was\ the\ best\ of)$$

Why isn't this practical?

SIMPLIFYING ASSUMPTION

- We can simplify this with the **Markov Assumption**
- We will only use the previous k words instead of the entire context
- For example,

$$P(times \mid best\ of) \approx P(times \mid it\ was\ the\ best\ of)$$

- Or generally,

$$P(w_1 w_2 \ldots w_n) \approx \prod_i^n P(w_i \mid w_{i-k} \ldots w_{i-1})$$

N-GRAM MODELS USE A LIMITED HISTORY

- Unigram – $P(w_1)$; no history
- Bigram – $P(w_2|w_1)$; 1 word as history
- Trigram – $P(w_3|w_1w_2)$; 2 words as history
- N-gram – $P(w_n|w_1w_2\ldots w_{n-1})$; n-1 words as history

- When would you want to use one or the other?

ESTIMATING N-GRAMS

With the **Maximum Likelihood Estimate** (MLE):

$$P(w_i|w_{i-1}) = \frac{count(w_{i-1}, w_i)}{count(w_{i-1})}$$

MLE EXAMPLE

$$P(w_i|w_{i-1}) = \frac{count(w_{i-1}, w_i)}{count(w_{i-1})}$$

<s> I am Sam </s>
<s> Sam I am </s>
<s> I do not like green eggs and ham </s>

$$P(\text{I} \mid \text{<s>}) = \tfrac{2}{3} = .67 \qquad P(\text{Sam} \mid \text{<s>}) = \tfrac{1}{3} = .33 \qquad P(\text{am} \mid \text{I}) = \tfrac{2}{3} = .67$$

$$P(\text{</s>} \mid \text{Sam}) = \tfrac{1}{2} = 0.5 \qquad P(\text{Sam} \mid \text{am}) = \tfrac{1}{2} = .5 \qquad P(\text{do} \mid \text{I}) = \tfrac{1}{3} = .33$$

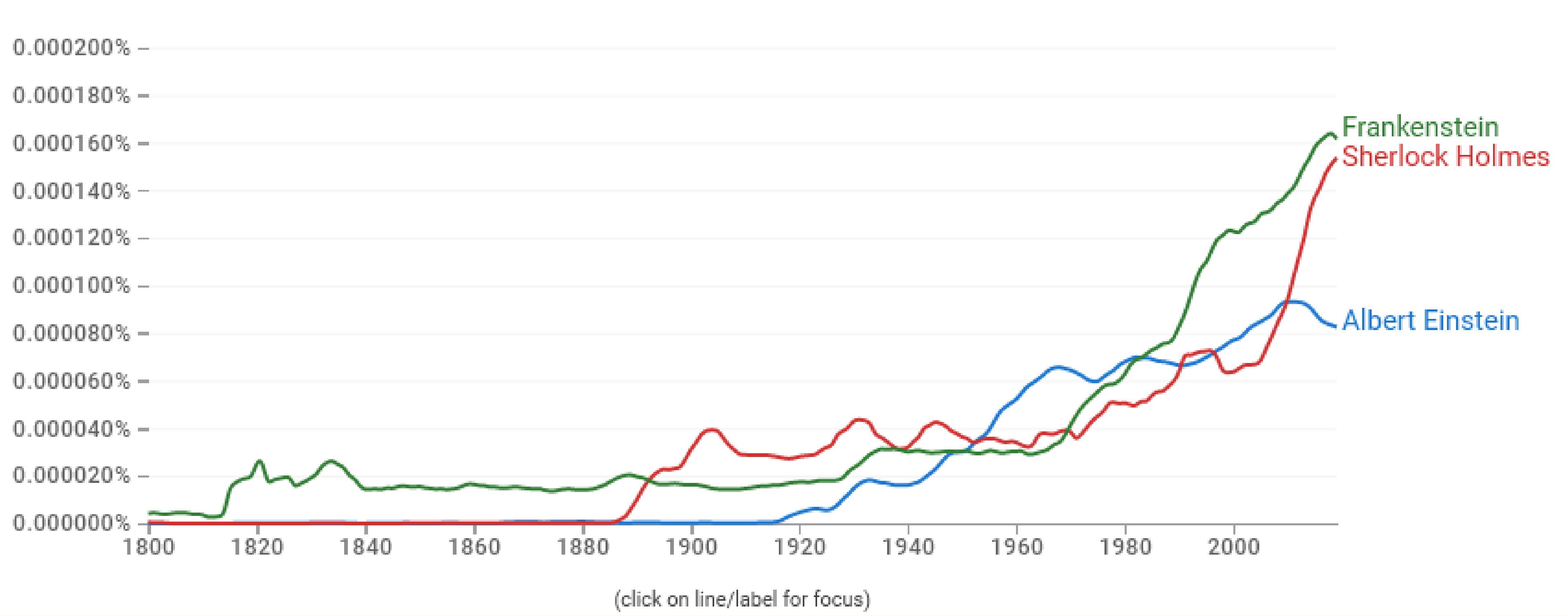
Example from “Speech and Language Processing”, Chapter 3 by Jurafsky & Martin

MLE EXTENDED TO MORE N-GRAMS

unigram	**no history**	$\prod_i^n p(w_i)$	$p(w_i) = \frac{count(w_i)}{all\ words}$
bigram	1 word as history	$\prod_i^n p(w_i \mid w_{i-1})$	$p(w_i \mid w_{i-1}) = \frac{count(w_{i-1}, w_i)}{count(w_{i-1})}$
trigram	2 words as history	$\prod_i^n p(w_i \mid w_{i-2}, w_{i-1})$	$p(w_i \mid w_{i-2}, w_{i-1}) = \frac{count(w_{i-2}, w_{i-1}, w_i)}{count(w_{i-2}, w_{i-1})}$
4-gram	3 words as history	$\prod_i^n p(w_i \mid w_{i-3}, w_{i-2}, w_{i-1})$	$p(w_i \mid w_{i-3}, w_{i-2}, w_{i-1}) = \frac{count(w_{i-3}, w_{i-2}, w_{i-1}, w_i)}{count(w_{i-3}, w_{i-2}, w_{i-1})}$

FUN WITH N-GRAMS

Check out https://books.google.com/ngrams/ for an interactive view of n-grams



ISSUES WITH N-GRAMS

- Long-distance dependencies
  - E.g. The **picture is** beautiful. vs The **picture** of the Sicilian landscape **is** beautiful.
- No idea what the probability of novel words would be
  - Misspellings will be counted as separate words
- If it's not found in dataset used to create the n-grams, there is no data on it
- Word disambiguation – same word with different meanings in different contexts

ISSUES WITH MLE

- Zeroes – dataset too small or doesn't match what we want the probability for (Out of vocabulary)
  - E.g.,

Train	Test
denied the allegations	denied the memo
denied the reports	
denied the claims	
denied the requests	

P(memo | denied the) = 0
And we also assign 0 probability to all sentences containing it!

- Solution: Smoothing