



Applied Natural Language Processing

Info 256

Lecture 11: Language models 1 (Oct 2, 2023)

David Bamman, UC Berkeley

Language Model

- Vocabulary \mathcal{V} is a finite set of discrete symbols (e.g., words, characters); $V = |\mathcal{V}|$
- \mathcal{V}^+ is the infinite set of sequences of symbols from \mathcal{V} ; each sequence ends with **STOP**
- $x \in \mathcal{V}^+$

Language Model

$$P(w) = P(w_1, \dots, w_n)$$

$$P(\text{"Call me Ishmael"}) = \\ P(w_1 = \text{"call"}, w_2 = \text{"me"}, w_3 = \text{"Ishmael"}) \times P(\text{STOP})$$

$$\sum_{w \in V^+} P(w) = 1$$

$$0 \leq P(w) \leq 1$$

over all sequence lengths!

Language Model

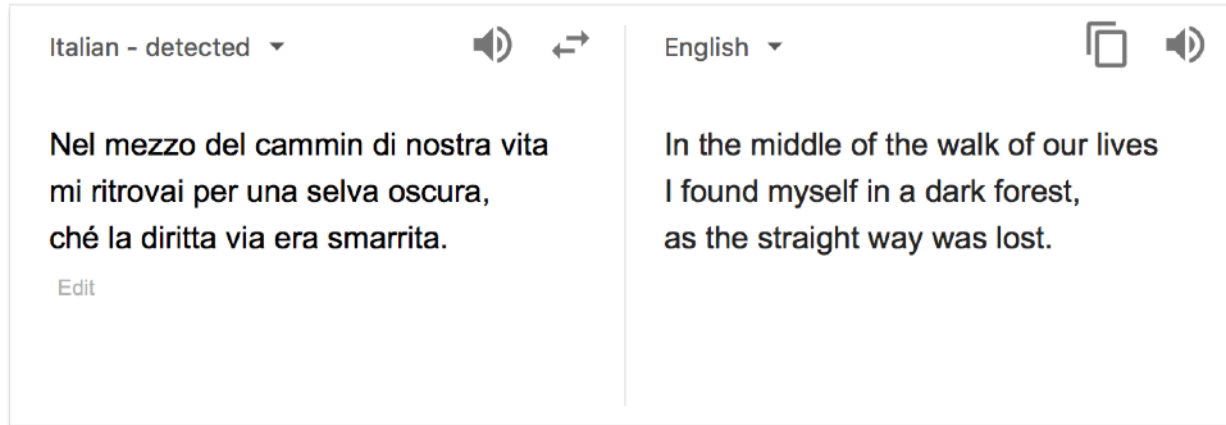
- Language models provide us with a way to quantify the likelihood of a sequence — i.e., **plausible** sentences.

OCR



To see great *Pompey* passe the streets of Rome :
And when you saw his Chariot but appeare,
Haue you not made an Vniuersall shout,
That Tyber trembled vnderneath her bankes
To heare the replication of your sounds,
Made in her Concaue Shores?

- to see great Pompey passe the streets of Rome:
- to see great Pompey passe the streets of Rome:

Machine translation





The screenshot shows a machine translation interface with two panels. The left panel is for the source text in Italian, and the right panel is for the translated text in English. Both panels include a speaker icon for audio playback and a bidirectional arrow icon for switching languages. The Italian text is: "Nel mezzo del cammin di nostra vita mi ritrovai per una selva oscura, ché la diritta via era smarrita." Below it is an "Edit" link. The English translation is: "In the middle of the walk of our lives I found myself in a dark forest, as the straight way was lost."

Italian - detected ▾  

Nel mezzo del cammin di nostra vita
mi ritrovai per una selva oscura,
ché la diritta via era smarrita.

[Edit](#)

English ▾  

In the middle of the walk of our lives
I found myself in a dark forest,
as the straight way was lost.

- Fidelity (to source text)
- Fluency (of the translation)



natural lan

natural language processing

natural language understanding

natural language processing with python

natural language generation

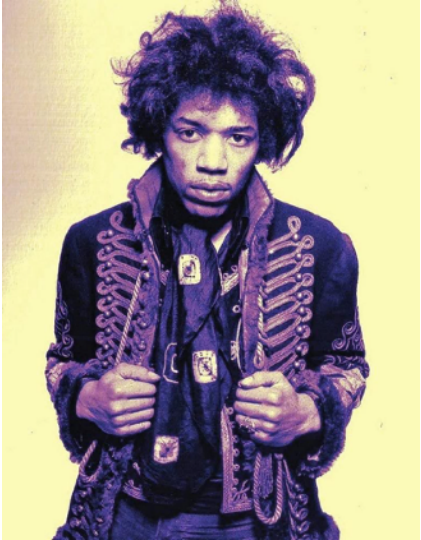
Google Search

I'm Feeling Lucky



Report inappropriate predictions

Speech Recognition



- 'Scuse me while I kiss the sky.
- 'Scuse me while I kiss this guy
- 'Scuse me while I kiss this fly.
- 'Scuse me while my biscuits fry

Dialogue generation

Q: What is your favorite animal?

A: My favorite animal is a dog.

Q: Why?

A: Because dogs are loyal and friendly.

Q: What are two reasons that a dog might be in a bad mood?

A: Two reasons that a dog might be in a bad mood are if it is hungry or if it is hot.

Q: How many bonks are in a quoit?

A: There are three bonks in a quoit.

Q: How many rainbows does it take to jump from Hawaii to seventeen?

A: It takes two rainbows to jump from Hawaii to seventeen.

Language Model

- Language modeling is the task of estimating $P(w)$
- Why is this hard?

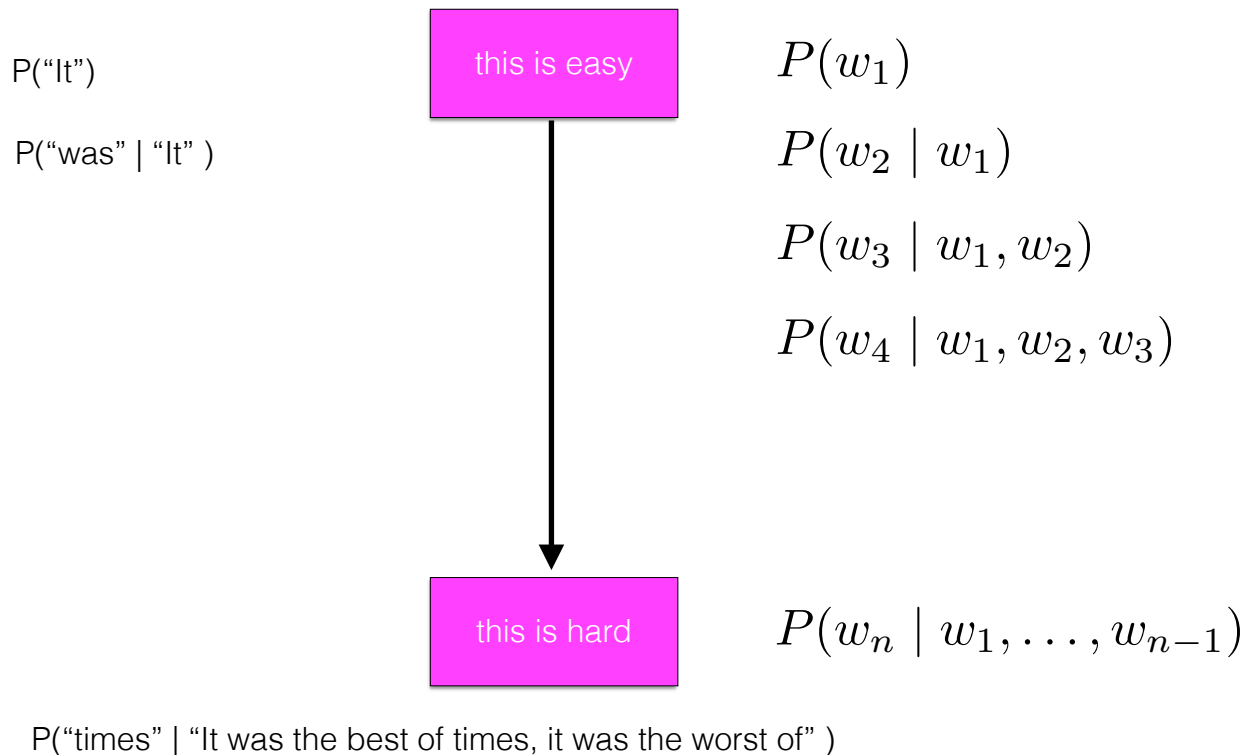
$P(\text{"It was the best of times, it was the worst of times"})$

Chain rule (of probability)

$$\begin{aligned} P(x_1, x_2, x_3, x_4, x_5) &= P(x_1) \\ &\times P(x_2 \mid x_1) \\ &\times P(x_3 \mid x_1, x_2) \\ &\times P(x_4 \mid x_1, x_2, x_3) \\ &\times P(x_5 \mid x_1, x_2, x_3, x_4) \end{aligned}$$

P("It was the best of times, it was the worst of times")

Chain rule (of probability)



Markov assumption

first-order

$$P(x_i \mid x_1, \dots, x_{i-1}) \approx P(x_i \mid x_{i-1})$$

second-order

$$P(x_i \mid x_1, \dots, x_{i-1}) \approx P(x_i \mid x_{i-2}, x_{i-1})$$

Markov assumption

bigram model
(first-order markov)

$$\prod_i^n P(w_i | w_{i-1}) \times P(\text{STOP} | w_n)$$

trigram model
(second-order markov)

$$\prod_i^n P(w_i | w_{i-2}, w_{i-1}) \\ \times P(\text{STOP} | w_{n-1}, w_n)$$

$$P(\textit{It} \mid \text{START}_1, \text{START}_2)$$

$$P(\textit{was} \mid \text{START}_2, \textit{It})$$

$$P(\textit{the} \mid \textit{It}, \textit{was})$$

“It was the best of
times, it was the
worst of times”

...

$$P(\textit{times} \mid \textit{worst}, \textit{of})$$

$$P(\text{STOP} \mid \textit{of}, \textit{times})$$

Estimation

unigram

$$\prod_i^n P(w_i) \\ \times P(STOP)$$

bigram

$$\prod_i^n P(w_i | w_{i-1}) \\ \times P(STOP | w_n)$$

trigram

$$\prod_i^n P(w_i | w_{i-2}, w_{i-1}) \\ \times P(STOP | w_{n-1}, w_n)$$

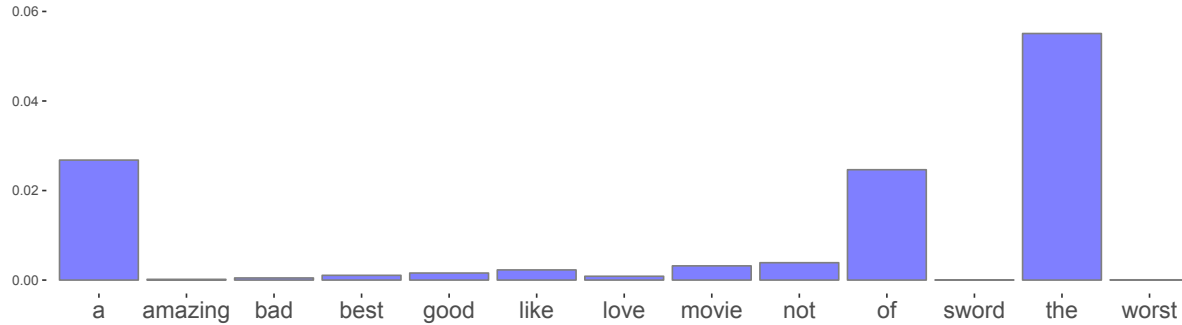
Maximum likelihood estimate

$$\frac{c(w_i)}{N}$$

$$\frac{c(w_{i-1}, w_i)}{c(w_{i-1})}$$

$$\frac{c(w_{i-2}, w_{i-1}, w_i)}{c(w_{i-2}, w_{i-1})}$$

Generating



- What we learn in estimating language models is $P(\text{word} \mid \text{context})$, where context — at least here — is the previous n words (for ngram of order n)
- We have one multinomial over the vocabulary (including **STOP**) for each context

Generating

- As we sample, the words we generate form the new context we condition on

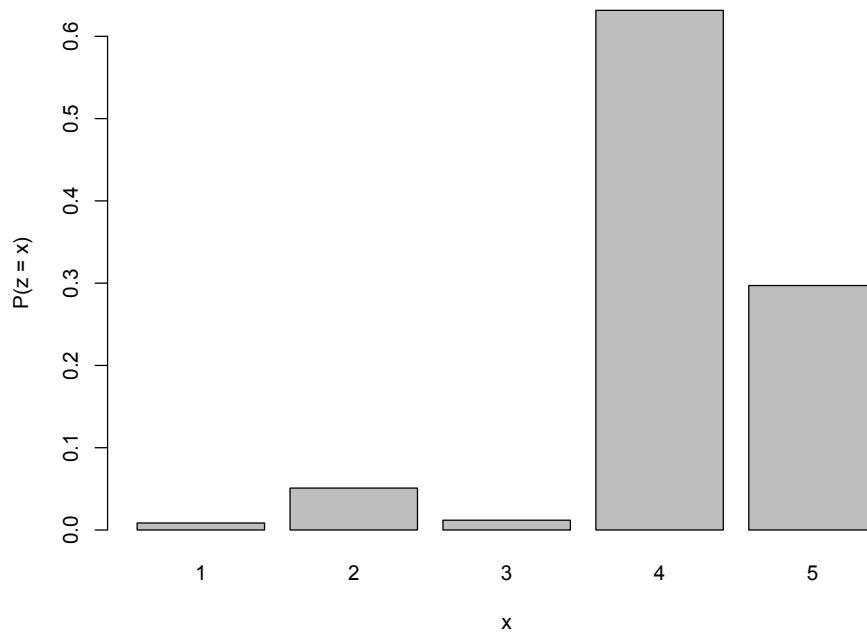
context1	context2	generated word
START	START	The
START	The	dog
The	dog	walked
dog	walked	in

Aside: sampling?

Sampling from a Multinomial

Probability
mass function
(PMF)

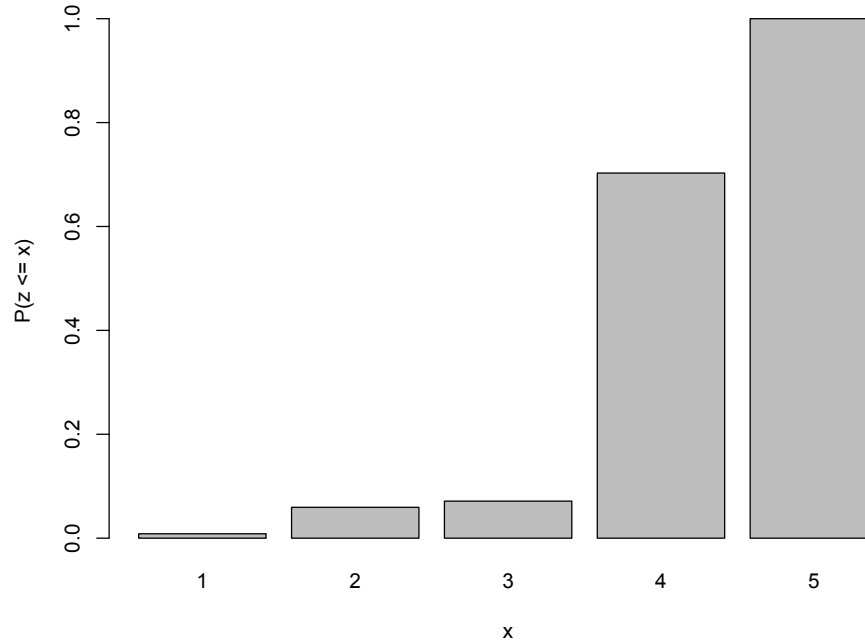
$P(z = x)$ exactly



Sampling from a Multinomial

Cumulative
density
function (CDF)

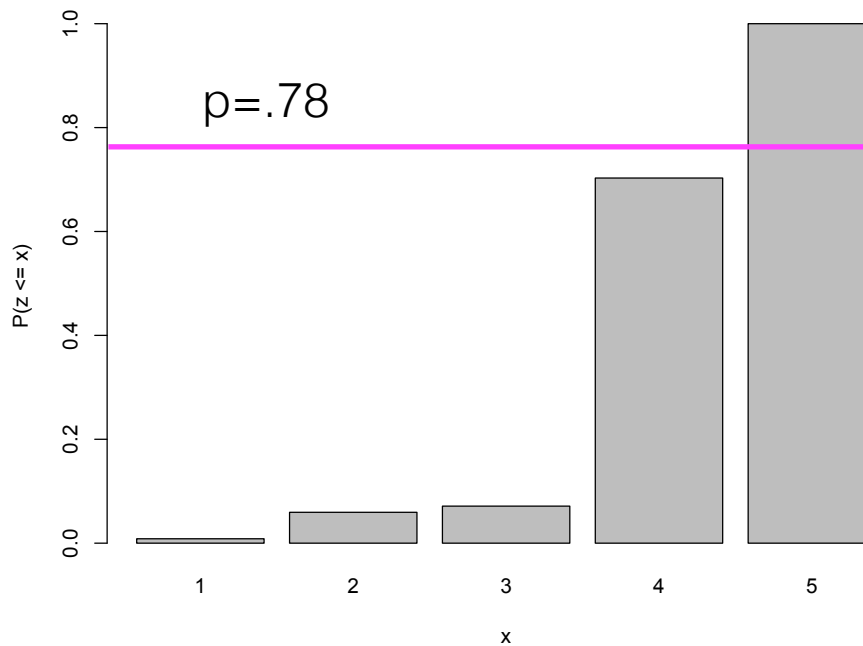
$$P(z \leq x)$$



Sampling from a Multinomial

Sample p
uniformly in
 $[0,1]$

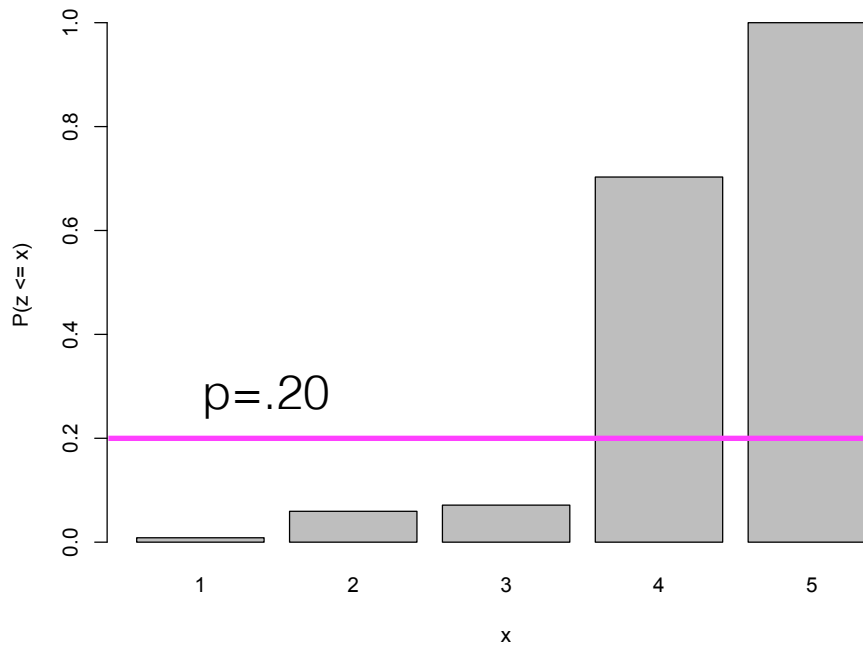
Find the point
 $\text{CDF}^{-1}(p)$



Sampling from a Multinomial

Sample p
uniformly in
 $[0,1]$

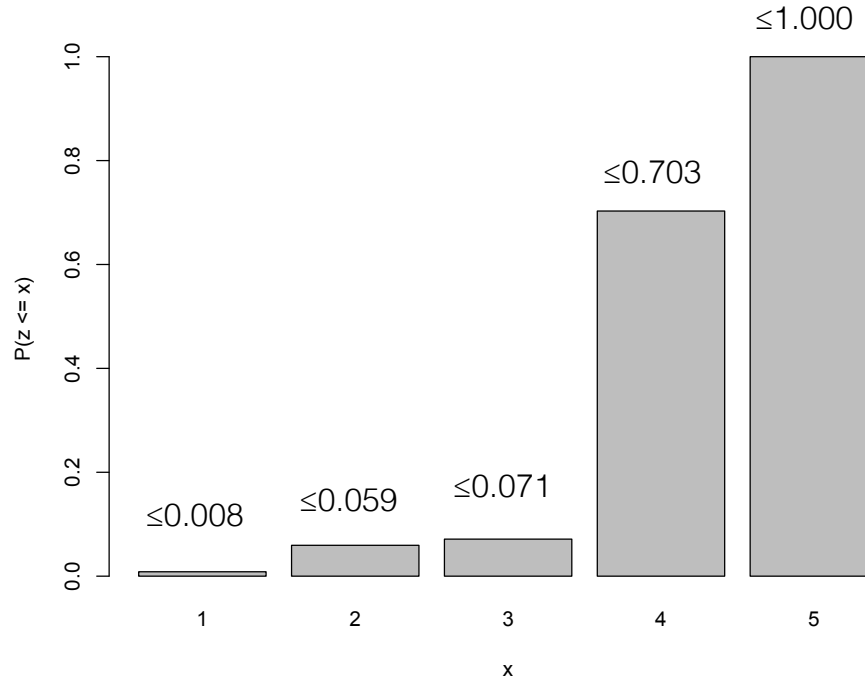
Find the point
 $\text{CDF}^{-1}(p)$



Sampling from a Multinomial

Sample p
uniformly in
 $[0,1]$

Find the point
 $\text{CDF}^{-1}(p)$



Unigram model

- the around, she They I blue talking “Don’t to and little come of
- on fallen used there. young people to Lázaro
- of the
- the of of never that ordered don't avoided to complaining.
- words do had men flung killed gift the one of but thing seen I plate
Bradley was by small Kingmaker.

Bigram Model

- “What the way to feel where we’re all those ancients called me one of the Council member, and smelled Tales of like a Korps peaks.”
- Tuna battle which sold or a monocle, I planned to help and distinctly.
- “I lay in the canoe ”
- She started to be able to the blundering collapsed.
- “Fine.”

Trigram Model

- “I’ll worry about it.”
- Avenue Great-Grandfather Edgeworth hasn’t gotten there.
- “If you know what. It was a photograph of seventeenth-century flourishin’ To their right hands to the fish who would not care at all. Looking at the clock, ticking away like electronic warnings about wonderfully SAT ON FIFTH
- Democratic Convention in rags soaked and my past life, I managed to wring your neck a boss won’t so David Pritchett giggled.
- He humped an argument but her bare He stood next to Larry, these days it will have no trouble Jay Grayer continued to peer around the Germans weren’t going to faint in the

4gram Model

- Our visitor in an idiot sister shall be blotted out in bars and flirting with curly black hair right marble, wallpapered on screen credit.”
- You are much instant coffee ranges of hills.
- Madison might be stored here and tell everyone about was tight in her pained face was an old enemy, trading-posts of the outdoors watching Anyog extended On my lips moved feebly.
- said.
- “I’m in my mind, threw dirt in an inch,’ the Director.

Evaluation

- The best evaluation metrics are **external** — how does a better language model influence the application you care about?
- Speech recognition (word error rate), machine translation (BLEU score), topic models (sensemaking)

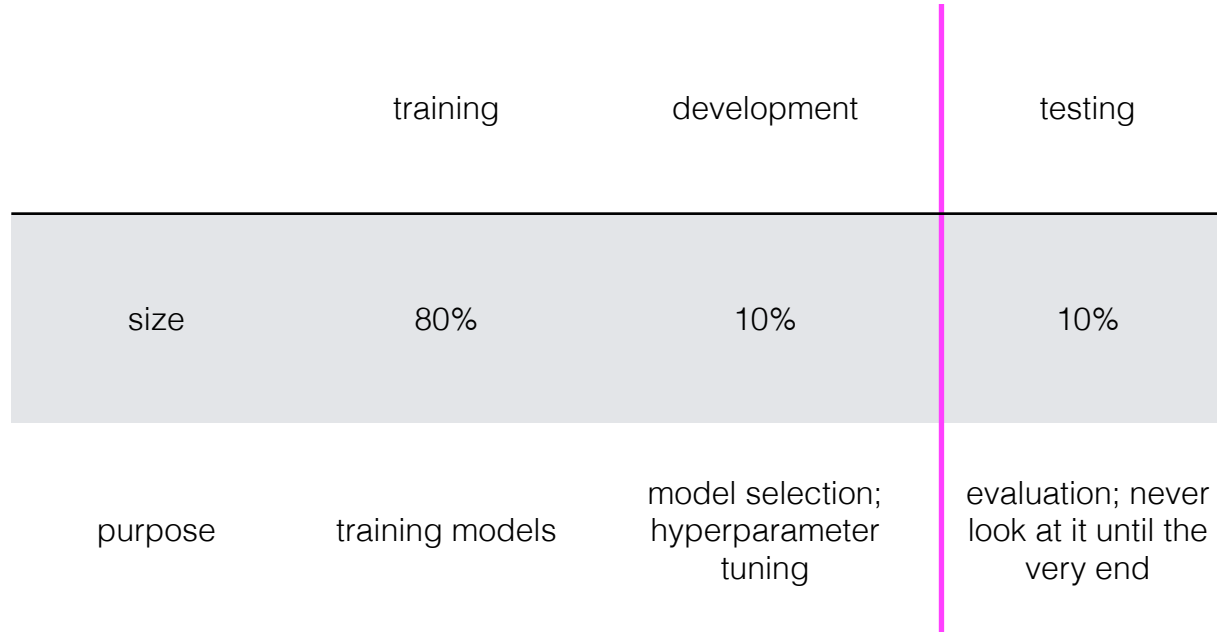
Evaluation

- A good language model should judge **unseen real language** to have high probability
- Perplexity = inverse probability of test data, averaged by word.
- To be reliable, the test data must be truly unseen (including knowledge of its vocabulary).

$$\text{perplexity} = \sqrt[N]{\frac{1}{P(w_1, \dots, w_n)}}$$

$$\begin{aligned}\sqrt[N]{\frac{1}{\prod_i^N P(w_i)}} &= \left(\prod_i^N P(w_i) \right)^{-\frac{1}{N}} \\ &= \exp \log \left(\prod_i^N P(w_i) \right)^{-\frac{1}{N}} \\ &= \exp \left(-\frac{1}{N} \log \prod_i^N P(w_i) \right) \\ \text{perplexity} &= \exp \left(-\frac{1}{N} \sum_i^N \log P(w_i) \right)\end{aligned}$$

Experiment design



Perplexity

bigram model
(first-order markov)

$$= \exp \left(-\frac{1}{N} \sum_i^N \log P(w_i | w_{i-1}) \right)$$

trigram model
(second-order markov)

$$= \exp \left(-\frac{1}{N} \sum_i^N \log P(w_i | w_{i-2}, w_{i-1}) \right)$$

Perplexity

Model	Unigram	Bigram	Trigram
Perplexity	962	170	109

SLP3 4.3

Smoothing

- When estimating a language model, we're relying on the data we've observed in a **training corpus**.
- Training data is a small (and biased) sample of the **creativity** of language.

Data sparsity

	i	want	to	eat	chinese	food	lunch	spend
i	5	827	0	9	0	0	0	2
want	2	0	608	1	6	6	5	1
to	2	0	4	686	2	0	6	211
eat	0	0	2	0	16	2	42	0
chinese	1	0	0	0	0	82	1	0
food	15	0	15	0	1	4	0	0
lunch	2	0	0	0	0	1	0	0
spend	1	0	1	0	0	0	0	0

Figure 4.1 Bigram counts for eight of the words (out of $V = 1446$) in the Berkeley Restaurant Project corpus of 9332 sentences. Zero counts are in gray.

- $P(w_i) = 0$ causes $P(w) = 0$. (Perplexity?)

$$\text{perplexity} = \exp\left(-\frac{1}{N} \sum_i^N \log P(w_i)\right)$$

Smoothing

- One solution: add a little probability mass to every element.

maximum likelihood
estimate

$$P(x_i | y) = \frac{n_{i,y}}{n_y}$$

$n_{i,y}$ = count of word i following context y
 n_y = count of context y
 V = size of vocabulary

smoothed estimates

$$P(x_i | y) = \frac{n_{i,y} + a}{n_y + Va}$$

same a for all x_i

$$P(x_i | y) = \frac{n_{i,y} + a_i}{n_y + \sum_{j=1}^V a_j}$$

possibly different a for each x_i



Classification

A mapping h from input data x (drawn from instance space \mathcal{X}) to a label (or labels) y from some enumerable output space \mathcal{Y}

\mathcal{X} = set of all documents

\mathcal{Y} = {english, mandarin, greek, ...}

x = a single document

y = ancient greek



Classification

A mapping h from input data x (drawn from instance space \mathcal{X}) to a label (or labels) y from some enumerable output space \mathcal{Y}

$\mathcal{Y} = \{\text{the, of, a, dog, iphone, ...}\}$

$x = (\text{context})$

$y = \text{word}$

x

y

In an attempt to modernize how visitors experience its 19th-century building, the Metropolitan Museum of Art is planning to turn the large store off its Great Hall into an 11,500-square-foot gallery for its blockbuster Costume Institute exhibitions and to transform an entrance underneath the main staircase into a retail space and restaurant that will be open to the public even when the museum is _____

closed

Multiclass logistic regression

$$P(Y = y | X = x; \beta) = \frac{\exp(x^\top \beta_y)}{\sum_{y' \in \mathcal{Y}} \exp(x^\top \beta_{y'})}$$

output space

$$\mathcal{Y} = \{1, \dots, K\}$$

Language Model

- We can use multiclass logistic regression for language modeling by treating the vocabulary as the output space

$$\mathcal{Y} = \mathcal{V}$$

x

y

In an attempt to modernize how visitors experience its 19th-century building, the Metropolitan Museum of Art is planning to turn the large store off its Great Hall into an 11,500-square-foot gallery for its blockbuster Costume Institute exhibitions and to transform an entrance underneath the main staircase into a retail space and restaurant that will be open to the public even when the museum is _____.

closed

In

x

y

an

x

y

In an

attempt

x

y

In an attempt

to

x

In an attempt to

y

modernize

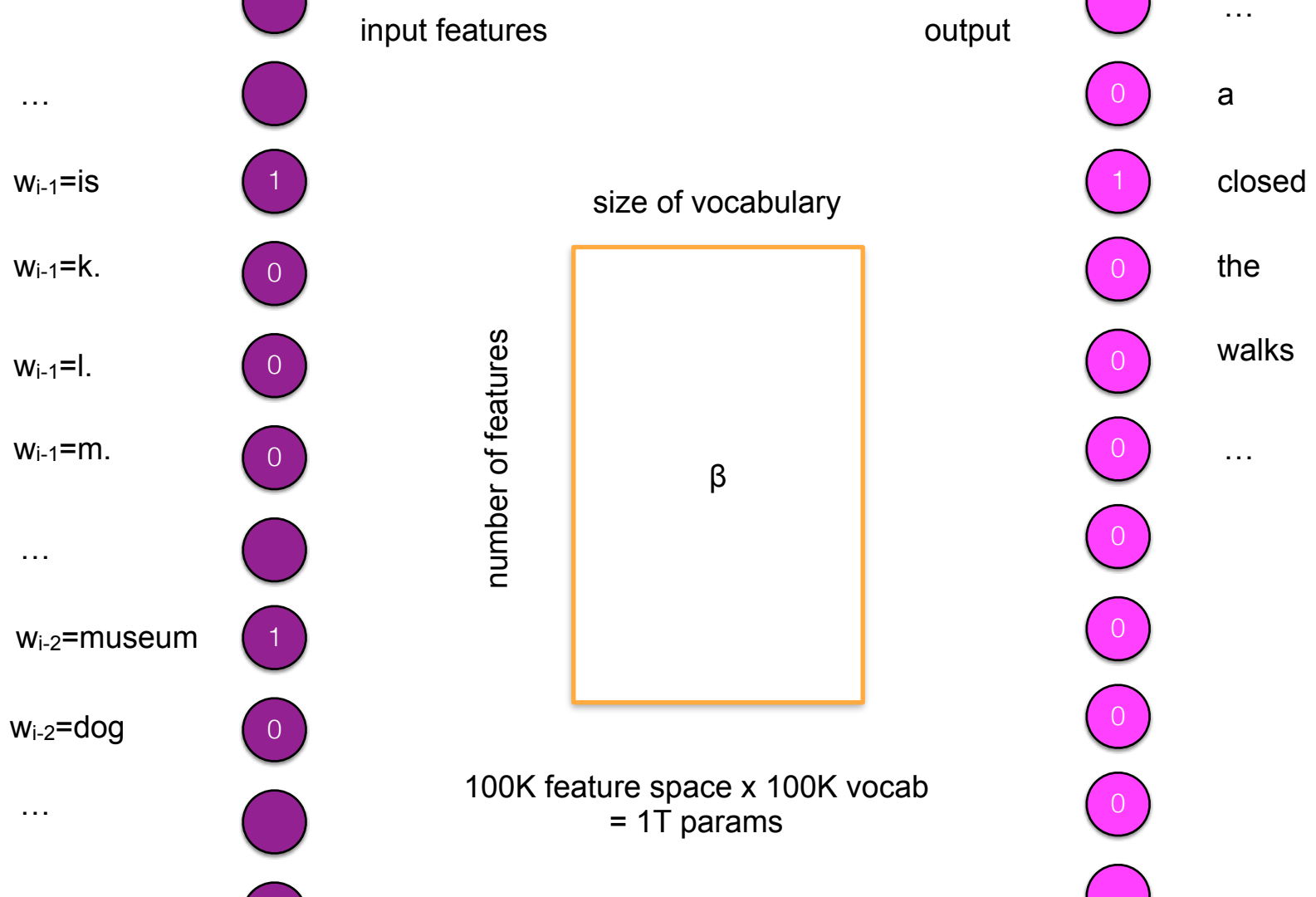
Richer representations

- Log-linear models give us the flexibility of encoding richer representations of the **context** we are conditioning on.
- We can reason about any observations from the entire history and not just the local context.

Tradeoffs

- Richer representations = more parameters, higher likelihood of overfitting
- Much slower to train than estimating the parameters of a classical model

$$P(Y = y \mid X = x; \beta) = \frac{\exp(x^\top \beta_y)}{\sum_{y' \in \mathcal{Y}} \exp(x^\top \beta_{y'})}$$



Activity

`8.lm/ExploreLM.ipynb`