

Natural Language Processing

Info 159/259

Lecture 7: Language models 1 (Feb 7, 2024)

*Many slides & instruction ideas borrowed from:
David Bamman, Mohit Iyyer & Sofia Serrano*

Logistics

- Quiz 3 will be out this Friday (due next Monday Feb 12).
- Homework 2 is out & due next Tuesday, Feb 13 (11:59 pm)
 - Homework 3 will be out mid next week.
- Annotation Project starting
 - AP0 will be out soon.
- Today:
 - N-gram Language Models
 - Annotation Project

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**
- $w \in \mathcal{V}^+$

Language Model

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

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

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

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

over all sequence lengths!

Language Model

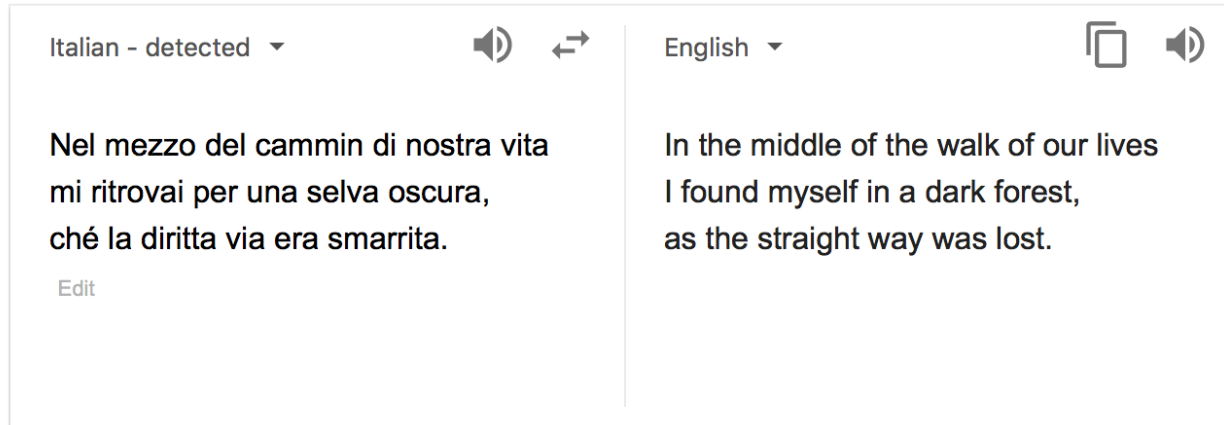
- Language models provide us with a way to quantify the likelihood of a sequence — i.e., **plausible/reasonable** sentences for the specific language.

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 columns. The left column is for the source text in Italian, and the right column is for the translated text in English. Both columns have a speaker icon for audio playback and a bidirectional arrow icon for switching directions. 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.

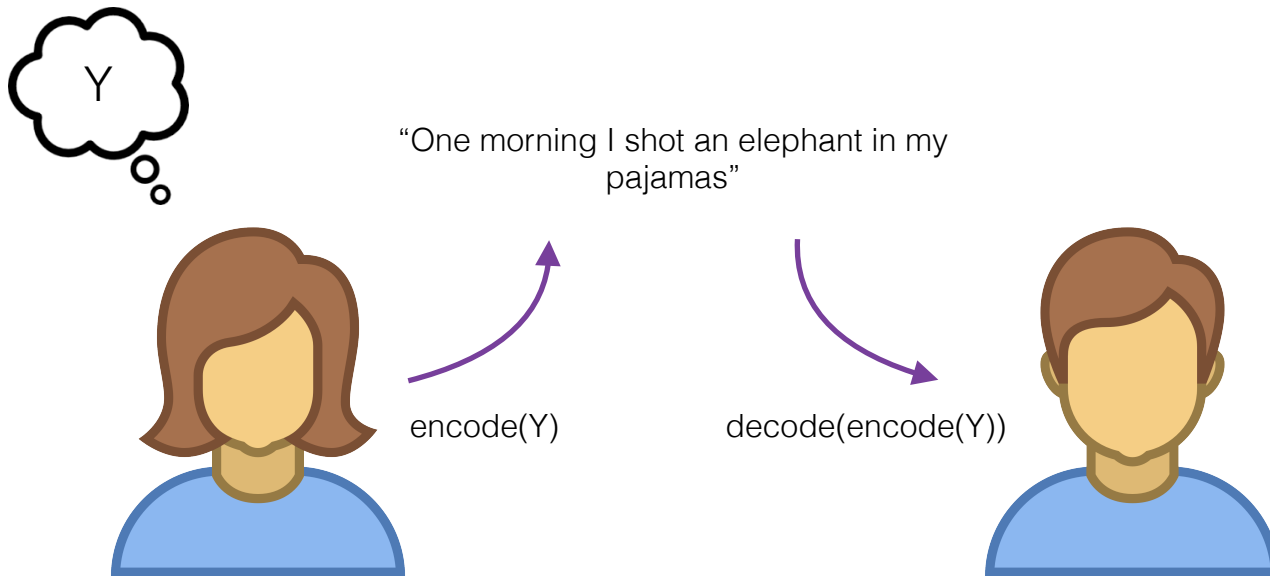
Q: How many bonks are in a quoit?

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.

Information theoretic view



Noisy Channel

	X	Y
ASR	speech signal	transcription
MT	source text	target text
OCR	pixel densities	transcription

$$P(Y | X) \propto \underbrace{P(X | Y)}_{\text{channel model}} \underbrace{P(Y)}_{\text{source model}}$$

Language Model

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

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

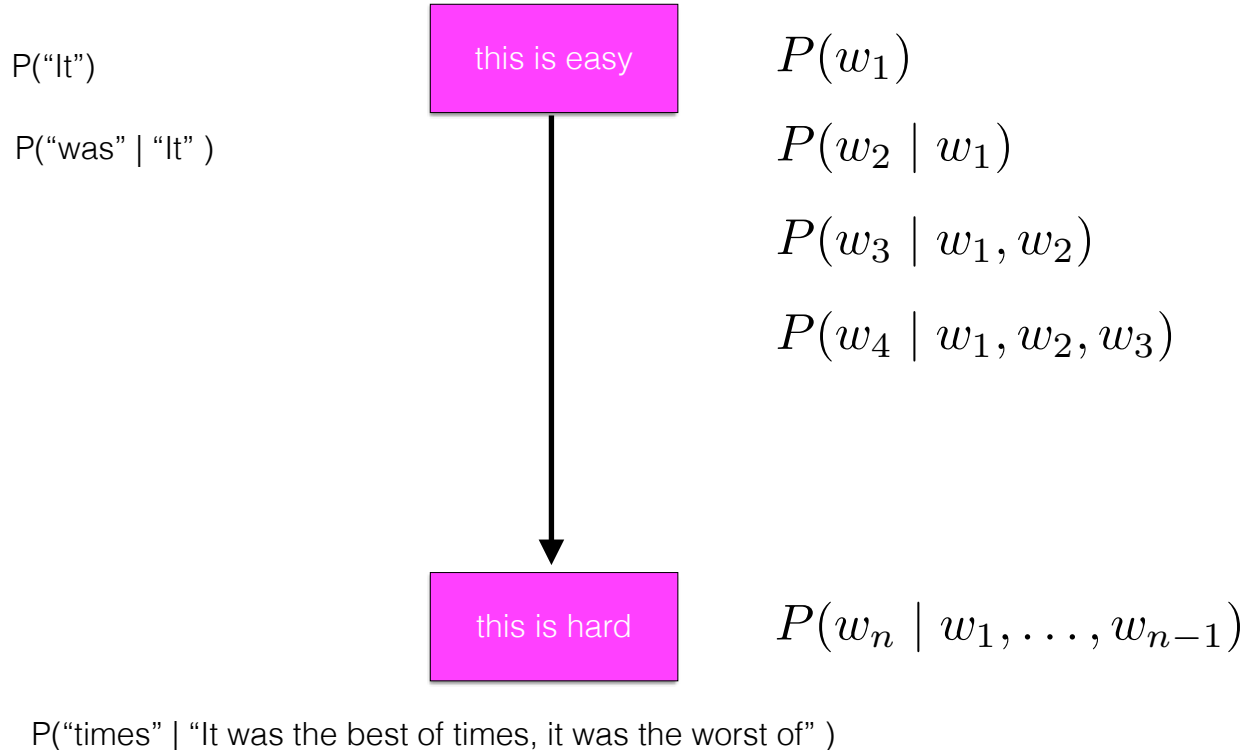
Chain rule (of probability)

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

Chain rule (of probability)

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

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

Ngram Models

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 of N-gram model

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

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

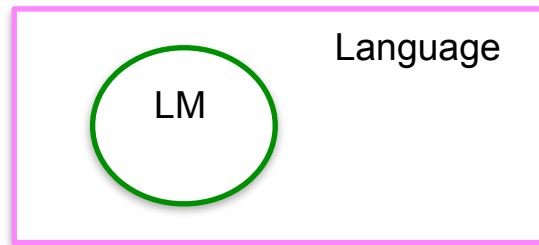
$$\frac{c(w_{i-1}, w_i)}{c(w_{i-1})}$$

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

Smoothing

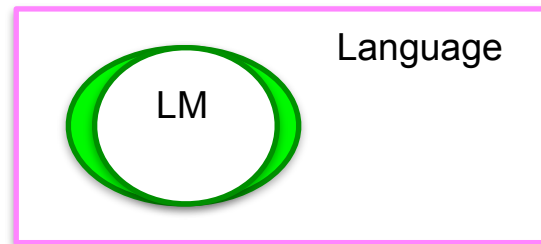
$$P(w_i) = \frac{c_i}{N}$$

- What about unseen ngrams?
- **Smoothing** reserves some probability mass for unseen events where each unseen ngram gets a tiny probability value.



Smoothing

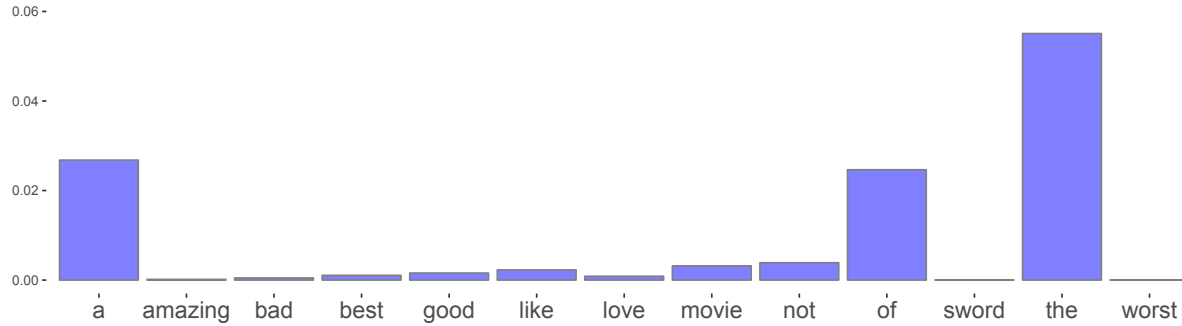
- **Smoothing** reserves some probability mass for unseen events where each unseen ngram gets a tiny probability value.



$$P(w_i) = \frac{c_i}{N}$$

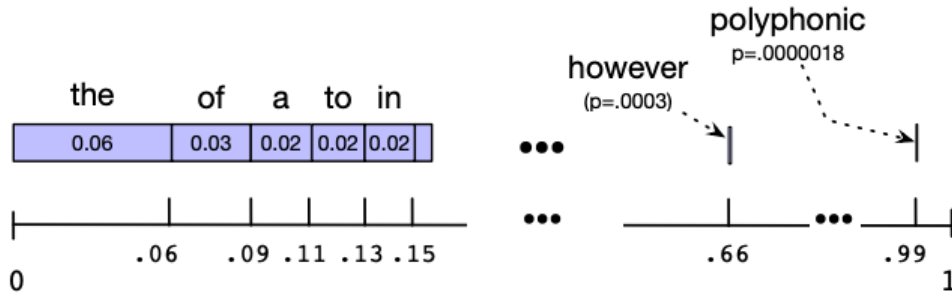
$$P_{\text{Laplace}}(w_i) = \frac{c_i + 1}{N + V}$$

Generating



- What we learn in estimating language models is $P(\text{word} \mid \text{context})$, where context — at least here — is the previous $n-1$ words (for ngram of order n)
- We have one multinomial over the vocabulary (including **STOP**) for each context

Generating via Sampling



- For sampling we choose a random number $x \in [0,1]$ and choose the word that the x belongs to its probability range.

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

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)

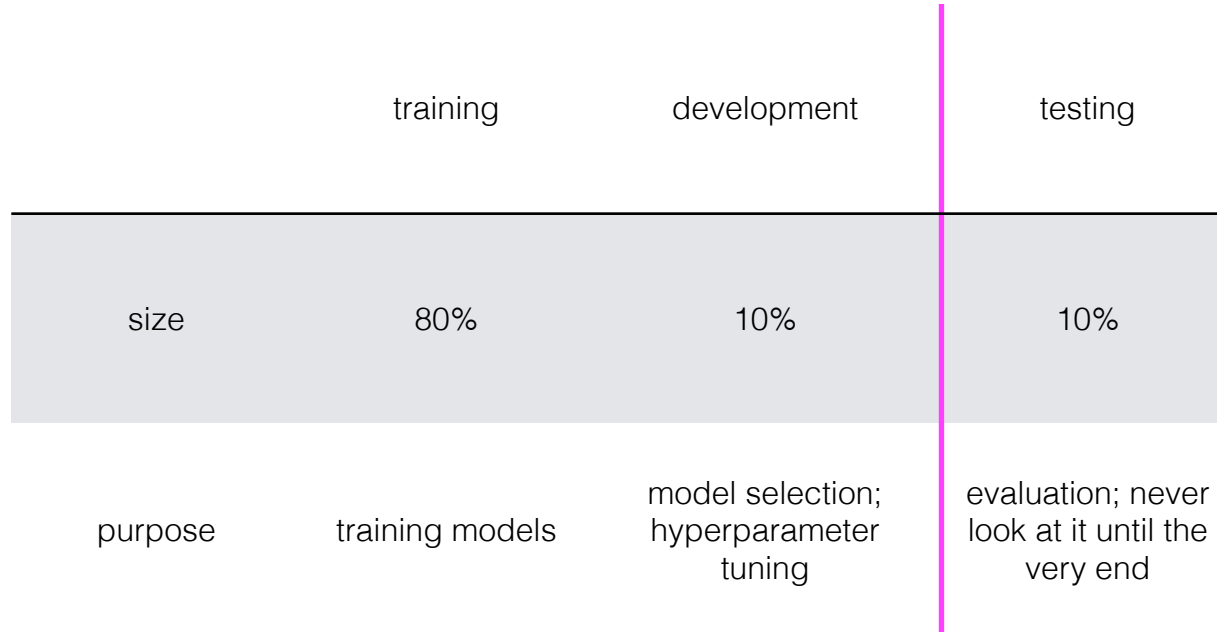
Evaluation

- 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

Interpolation

- As ngram order rises, we have the potential for higher **precision** but also higher **variability** in our estimates.
- A linear interpolation of any two language models p and q (with $\lambda \in [0,1]$) is also a valid language model.

$$\lambda p + (1 - \lambda)q$$

p = the web

q = political speeches

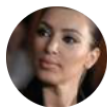
Interpolation

- We can use this fact to make higher-order language models more **robust**.

$$\begin{aligned} P(w_i \mid w_{i-2}, w_{i-1}) &= \lambda_1 P(w_i \mid w_{i-2}, w_{i-1}) \\ &\quad + \lambda_2 P(w_i \mid w_{i-1}) \\ &\quad + \lambda_3 P(w_i) \end{aligned}$$

$$\lambda_1 + \lambda_2 + \lambda_3 = 1$$

Mixture models



KimKierkegaardashian @KimKierkegaard · Sep 8



It is hard to try on clothes without some question arising as to my relationship to the eternal



1



387



822



KimKierkegaardashian @KimKierkegaard · Sep 8



The perfect white tee reminds you that being nothing in this world is the condition for being something in the next



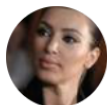
2



152



449



KimKierkegaardashian @KimKierkegaard · Sep 8



People despair about being lonely and therefore get married. But is this love? I should say it is self-love. Happy anniversary babe



1



136



549



Un **film** (in **Italiano** anche **pellicola** oppure in alcune parti d'Italia **cinema**), è un'opera d'**arte visiva** che simula esperienze e comunica in altro modo idee, storie, percezioni, sentimenti, bellezza o atmosfera attraverso l'uso di **immagini** in movimento.

بھی کہا (motion picture) یا متحرک تصویر (movie) جسے مووی (film) فلم جاتا ہے، ساکت تصاویر کا ایسا سلسلہ ہوتا ہے جو پردے (اسکرین) پر یوں دکھایا جاتا ہے کہ اس پر متحرک ہونے کا دھوکا ہوتا ہے۔ مختلف اشیاء کو تسلسل کے ساتھ تیز رفتاری سے دکھائے جانے کے باعث یہ بصری دھوکا ناظرین کو احساس دلاتا ہے کہ وہ مسلسل متحرک اشیاء دیکھ رہے ہیں۔ ایک **موشن پکچر کیمرے** کے ذریعے اصل مناظر کی عکس بندی کر کے فلم تخلیق کی جاتی ہے۔ **موشن پکچرے کیمرے** کے ذریعے اصل مناظر کی عکس بندی؛ تصاویر یا روایتی انیمیشن تکنیکیں استعمال کرتے ہوئے چھوٹی شبیہوں کی عکس بندی

Classification

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

\mathcal{X} = set of all documents

$\mathcal{Y} = \{it, ur, zh, en, es, ar, ..\}$

x = a single document

$y = it$

Classification

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

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

$x = (\text{context})$

$y = \text{word}$

Multi-class 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

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

x = feature vector

Feature	Value
the	0
and	0
bravest	0
love	0
loved	0
genius	0
not	0
fruit	1
<i>BIAS</i>	1

β = coefficients

Feature	β_1	β_2	β_3	β_4	β_5
the	1.33	-0.80	-0.54	0.87	0
and	1.21	-1.73	-1.57	-0.13	0
bravest	0.96	-0.05	0.24	0.81	0
love	1.49	0.53	1.01	0.64	0
loved	-0.52	-0.02	2.21	-2.53	0
genius	0.98	0.77	1.53	-0.95	0
not	-0.96	2.14	-0.71	0.43	0
fruit	0.59	-0.76	0.93	0.03	0
<i>BIAS</i>	-1.92	-0.70	0.94	-0.63	0

Language Model

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

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

Unigram LM

- A unigram language model here would have just one feature: a bias term.

Feature	β_{the}	β_{of}	β_{a}	β_{dog}	β_{iphone}
<i>BIAS</i>	-1.92	-0.70	0.94	-0.63	0

Bigram LM

Feature	Value
$w_{i-1}=\text{the}$	1
$w_{i-1}=\text{and}$	0
$w_{i-1}=\text{bravest}$	0
$w_{i-1}=\text{love}$	0
$w_{i-1}=\text{loved}$	0
$w_{i-1}=\text{dog}$	0
$w_{i-1}=\text{not}$	0
$w_{i-1}=\text{fruit}$	0
<i>BIAS</i>	1

β_{the}	β_{of}	β_{a}	β_{dog}	β_{iphone}
-0.78	-0.80	-0.54	0.87	0
1.21	-1.73	-1.57	-0.13	0
0.96	-0.05	0.24	0.81	0
1.49	0.53	1.01	0.64	0
-0.52	-0.02	2.21	-2.53	0
0.98	0.77	1.53	-0.95	0
-0.96	2.14	-0.71	0.43	0
0.59	-0.76	0.93	0.03	0
-1.92	-0.70	0.94	-0.63	0

$$P(w_i = \text{dog} \mid w_{i-1} = \text{the})$$

Feature	Value
$w_{i-1}=\text{the}$	1
$w_{i-1}=\text{and}$	0
$w_{i-1}=\text{bravest}$	0
$w_{i-1}=\text{love}$	0
$w_{i-1}=\text{loved}$	0
$w_{i-1}=\text{dog}$	0
$w_{i-1}=\text{not}$	0
$w_{i-1}=\text{fruit}$	0
<i>BIAS</i>	1

β_{the}	β_{of}	β_{a}	β_{dog}	β_{iphone}
-0.78	-0.80	-0.54	0.87	0
1.21	-1.73	-1.57	-0.13	0
0.96	-0.05	0.24	0.81	0
1.49	0.53	1.01	0.64	0
-0.52	-0.02	2.21	-2.53	0
0.98	0.77	1.53	-0.95	0
-0.96	2.14	-0.71	0.43	0
0.59	-0.76	0.93	0.03	0
-1.92	-0.70	0.94	-0.63	0

Trigram LM

$$P(w_i = \text{dog} \mid w_{i-2} = \text{and}, w_{i-1} = \text{the})$$

Feature	Value
$w_{i-2}=\text{the} \wedge w_{i-1}=\text{the}$	0
$w_{i-2}=\text{and} \wedge w_{i-1}=\text{the}$	1
$w_{i-2}=\text{bravest} \wedge w_{i-1}=\text{the}$	0
$w_{i-2}=\text{love} \wedge w_{i-1}=\text{the}$	0
$w_{i-2}=\text{loved} \wedge w_{i-1}=\text{the}$	0
$w_{i-2}=\text{genius} \wedge w_{i-1}=\text{the}$	0
$w_{i-2}=\text{not} \wedge w_{i-1}=\text{the}$	0
$w_{i-2}=\text{fruit} \wedge w_{i-1}=\text{the}$	0
<i>BIAS</i>	1

Smoothing and back-off

$$P(w_i = \text{dog} \mid w_{i-2} = \text{and}, w_{i-1} = \text{the})$$

second-order
features

first-order
features

Feature	Value
$w_{i-2}=\text{the} \wedge w_{i-1}=\text{the}$	0
$w_{i-2}=\text{and} \wedge w_{i-1}=\text{the}$	1
$w_{i-2}=\text{bravest} \wedge w_{i-1}=\text{the}$	0
$w_{i-2}=\text{love} \wedge w_{i-1}=\text{the}$	0
$w_{i-1}=\text{the}$	1
$w_{i-1}=\text{and}$	0
$w_{i-1}=\text{bravest}$	0
$w_{i-1}=\text{love}$	0
<i>BIAS</i>	1

L2 regularization

$$\ell(\beta) = \underbrace{\sum_{i=1}^N \log P(y_i | x_i, \beta)}_{\text{we want this to be high}} - \underbrace{\eta \sum_{j=1}^F \beta_j^2}_{\text{but we want this to be small}}$$

- We can do this by changing the function we're trying to optimize by adding a penalty for having values of β that are high
- η controls how much of a penalty to pay for coefficients that are far from 0 (optimize on development data)

L1 regularization

$$\ell(\beta) = \underbrace{\sum_{i=1}^N \log P(y_i | x_i, \beta)}_{\text{we want this to be high}} - \underbrace{\eta \sum_{j=1}^F |\beta_j|}_{\text{but we want this to be small}}$$

- L1 regularization encourages coefficients to be **exactly** 0.
- η again controls how much of a penalty to pay for coefficients that are far from 0 (optimize on development data)

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.

“JACKSONVILLE, Fla. — Stressed and exhausted families across the Southeast were assessing the damage from Hurricane Irma on Tuesday, even as flooding from the storm continued to plague some areas, like Jacksonville, and the worst of its wallop was being revealed in others, like the Florida Keys.

Officials in Florida, Georgia and South Carolina tried to prepare residents for the hardships of recovery from the _____”

The United States Senate opens its second impeachment trial of former President Donald J. _____

feature classes	example
ngrams ($w_{i-1}, w_{i-2}:w_{i-1}, w_{i-3}:w_{i-1}$)	w_{i-2} ="donald", w_{i-1} ="j."
gappy ngrams	w_1 ="impeachment" and w_2 ="donald"
spelling, capitalization	w_{i-1} is capitalized and w_i is capitalized
class/gazetteer membership	w_{i-1} in list of names and w_i in list of names

Tradeoffs

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

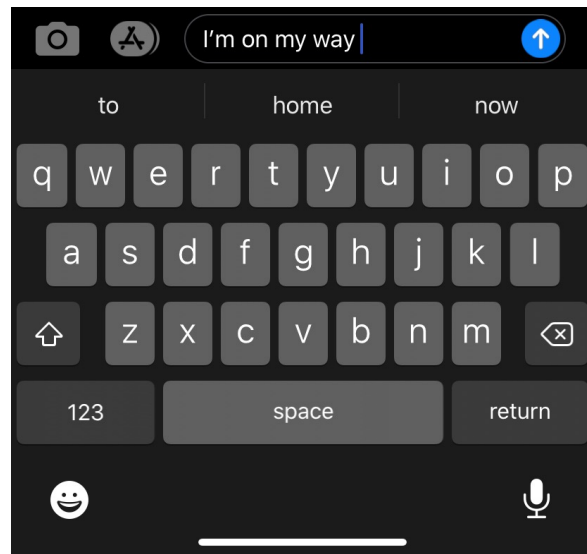
$$P(Y = y \mid X = x; \beta) = \frac{\exp(x^\top \beta_y)}{\sum_{y' \in \mathcal{Y}} \exp(x^\top \beta_{y'})}$$

Why?

- Language models give us an estimate for the probability of a sequence, which is directly useful for applications that are deciding between different sentences) as viable outputs:
 - Machine translation
 - Speech recognition
 - OCR
 - Dialogue agents

Why?

- Language models directly allow us to predict the next word in a sequence (useful for **autocomplete**).



Why?

- Language models can directly encode knowledge present in the training corpus.

The director of *2001: A Space Odyssey* is _____

Why?

- Language models can directly encode knowledge present in the training corpus.

Query	Answer	Generation
Francesco Bartolomeo Conti was born in ____.	Florence	Rome [-1.8], Florence [-1.8], Naples
Adolphe Adam died in ____.	Paris	Paris [-0.5], London [-3.5], Vienna
English bulldog is a subclass of ____.	dog	dogs [-0.3], breeds [-2.2], dog
The official language of Mauritius is ____.	English	English [-0.6], French [-0.9], Arabic
Patrick Oboya plays in ____ position.	midfielder	centre [-2.0], center [-2.2], midfielder
Hamburg Airport is named after ____.	Hamburg	Hess [-7.0], Hermann [-7.1], Schmidt

Why?

- Language modeling turns out to be a good proxy task for learning about linguistic structure.
- See contextual word embeddings (BERT/ELMo), in class 2/12.

Annotation project

- The most exciting applications of NLP have yet to be invented.
- Design a new NLP task and annotate data to support it, working in groups of **exactly 3 students**.

Existing tasks
Question answering
Named entity recognition
Sentiment analysis
Machine translation
Syntactic parsing
Coreference resolution
Text generation
Word sense disambiguation
...

AP deliverables

- **AP0.** Forming Teams
- **AP1.** Design a new document classification task and gather data to support it (must be shareable with the public — nothing private or in copyright).
- **AP2.** Create robust set of annotation guidelines, annotate the data, creating at least 500 labeled examples + reporting inter-annotator agreement rates.
- **AP3.** Build a classifier to automatically predict the labels using the data you've annotated.

Annotation Project

- Your annotation task must be one that requires **your human judgment** for the labels.
 - Some NLP tasks don't require human judgment — **don't** consider these!
 - stock price prediction (will \$GOOG go up or down on 2/1/22?) can use historical stock prices as labels.

Annotation Project

- You should manually be labeling your data, and not using algorithmic processes to do so (definitely no ChatGPT!).
 - Let's say you're annotating all of the mentions of **commercial products** in text (e.g., "Tide", "BMW", "Nintendo Switch") and you have a list of 5000 products that you're looking for, so you write some code to read in that list and then automatically tag all mentions of those 5000 products in the data. This is not interesting! (There is no need to label data for this in order to train a supervised system; you could simply run your algorithm instead.)

Annotation Project

- Your labels should not be deterministic, but really require some human comprehension of the context.
 - Let's say you're annotating how **suspenseful** a literary passage is, and every time a passage contains the words "thunder" you rate it a 3 for suspense; if it contains "gasped" you rate it 2; if it contains "anticipated" you rate it a 1; and 0 otherwise. If you can write an algorithm like this that can fully deterministically predict your gold labels correctly, then it's not interesting enough. (Again, there is no need to label data for this in order to train a supervised system; you could simply run your algorithm instead.)

Project Guideline

- <https://bcourses.berkeley.edu/courses/1531231/pages/annotation-project-guideline>
- Read the guideline carefully
- Team up
- AP0 will be out soon.

Logistics

- Quiz 3 will be out this Friday (due next Monday Feb 12).
- Homework 2 is out & due next Tuesday, Feb 13 (11:59 pm)
 - Homework 3 will be out mid next week.
- AP0 will be out soon.
- Next time:
 - Neural Language Models