LANGUAGE MODELS & N-GRAMS

Usman Shahid

Department of Computer Science

University of Illinois at Chicago

CS421 Natural Language Processing (Fall 2019)

Many slides adapted from Jurafsky and Martin

LANGUAGE MODELING TASK

Given

- A corpus of text
- **Corpus** (plural corpora): A computer readable collection of text or speech

Objective

- Build a statistical model that can assign probability to a sequence of "words" (or symbols)
- "words": tokens, types, lemmas, utterances, punctuations etc
- For a sequence of n words, compute $P(w_1, w_2, w_3, \dots, w_n)$
- $P(students, opened, the, book)$

ALTERNATE PERSPECTIVE

Given

- A corpus of text

Objective

- Build a statistical model which, given a sequence of words, can predict the probability of next word.
- Students opened the ________
- Predict nth word given n-1 words
  $P(w_n|w_1, w_2, w_3, \dots, w_{n-1})$
- $P(book|students, opened, the)$

CORPUS PERSPECTIVE

Given

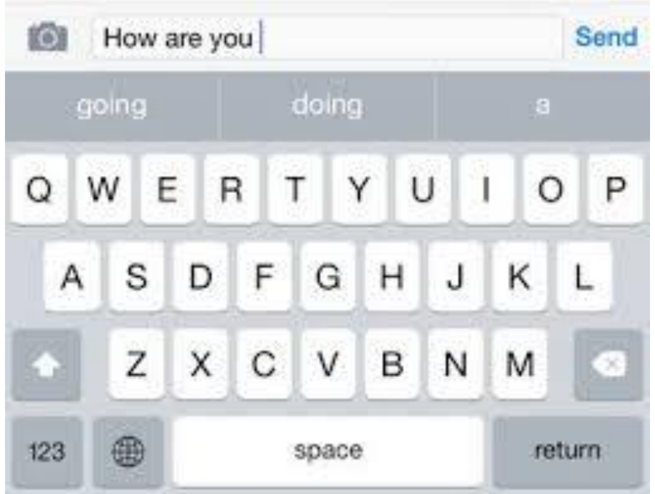
- A corpus of text

Objective

- Build a statistical model that assigns high probability to sentences that are “similar” to those in the corpus and vice verca
- Language Model trained on sports news articles
- Which sentence is more likely?
  - Real Madrid head back into La Liga action on Monday evening
  - Stage is set for record-breaking presidential primary debate

LANGUAGE MODEL: APPLICATIONS

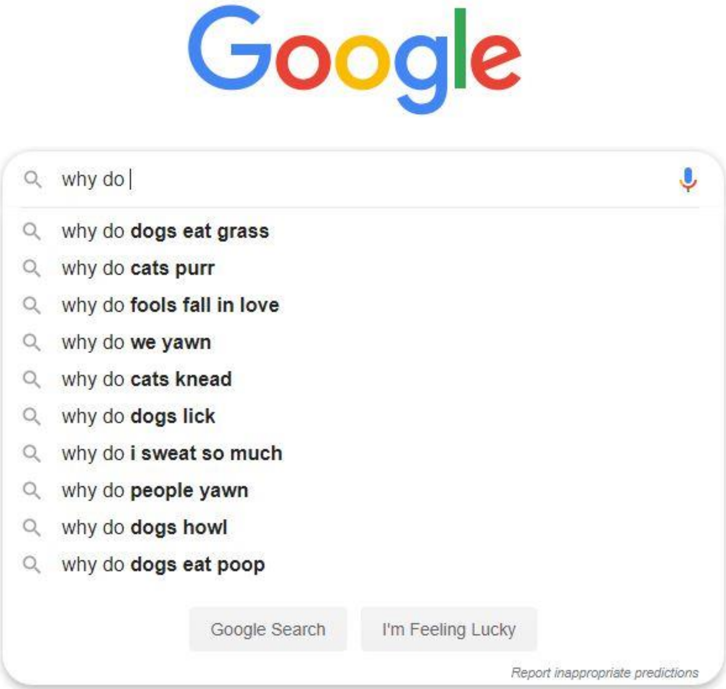
- Autocomplete


How are you

going

doing

a


why do

why do **dogs eat grass**

why do **cats purr**

why do **fools fall in love**

why do **we yawn**

why do **cats knead**

why do **dogs lick**

why do **i sweat so much**

why do **people yawn**

why do **dogs howl**

why do **dogs eat poop**

LANGUAGE MODEL: APPLICATIONS

- Machine Translation
  - Input
    - 他 (He) 向 (to) 记者 (reporters) 介绍了 (introduced) 主要 (main) 内容 (content)
  - Likely output
    - he introduced reporters to the main contents of the statement
    - he briefed to reporters the main contents of the statement
    - **he briefed reporters on the main contents of the statement**

LANGUAGE MODEL: APPLICATIONS

- Spelling correction
  - Their are two midterms
  - There are two midterms
- Speech recognition
  - I will be bassoon dish
  - I will be back soonish
- Authorship Attribution
  - How likely is this text written by Shakespeare?
  - How?
- And many more…

CHAIN RULE

- How do we compute probability of a sequence?
- Conditional probability
  - If, $P(B|A) = \frac{P(A,B)}{P(A)}$, then, $P(A,B) = P(A)P(B|A)$
- Chain rule
- $P(w_1, w_2, w_3, w_4) = P(w_1)P(w_2|w_1)P(w_3|w_1, w_2)\boldsymbol{P(w_4|w_1, w_2, w_3)}$
- $\boldsymbol{P(w_4|w_1, w_2, w_3)} = P(book|students, opened, the)$
- $P(w_1, w_2, \dots, w_{n-1}, w_n) = P(w_1)P(w_2|w_1) \dots P(w_n|w_1, w_2, \dots, w_{n-1})$
- $P(w_1, w_2, \dots, w_n) = \prod_i P(w_i|w_1, w_2, \dots, w_{i-1})$

LANGUAGE MODEL: CHALLENGING

- How to compute $P(book|students, opened, the)$
  - $\frac{Count(students\ opened\ the\ book)}{Count(students\ opened\ the)}$
- Issues?
  - Too many sentences
    - If sentence length = 4, vocab size = 100
    - 100^4 =100 million sentences!
  - Not enough data to estimate
    - In most cases $Count(students\ opened\ the\ book) = 0$

LANGUAGE MODEL: MARKOV ASSUMPTION

- Ngrams
  - Sequence of N consecutive words
  - Unigrams (1 word): students, opened, the, book
  - Bigrams (2 words): students opened, opened the, the book
  - Trigrams (3 words), students opened the, opened the book
  - 4-grams, 5-grams and so on..
- Markov Assumption
  - Consider only N previous words
  - Bigram model: $P(w_n|w_1, w_2 \ldots . w_{n-1}) \approx P(w_n|w_{n-1})$
    - $P(book|students, opened, the) \approx P(book|the)$
  - Trigram model: $P(w_n|w_1, w_2 \ldots . w_{n-1}) \approx P(w_n|w_{n-1}, w_{n-2})$
    - $P(book|students, opened, the) \approx P(book|opened, the)$

LANGUAGE MODEL: MLE

- Maximum Likelihood Estimation (MLE)
  - Set model parameters in such a way that the observed data is most probable
  - Model parameters
    - All N-gram probabilities
    - For example $P(book|the)$ for bigram language model
  - Estimate **relative frequencies**
    - Estimate parameters using counts
    - Normalize to get probabilities
- Bigram Counts
  - $P(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n)}{C(w_{n-1})}$
  - $P(book|the) = \frac{C(the\ book)}{C(the)}$

LANGUAGE MODEL: EXERCISE

- Corpus
  - <s> I am Sam </s>
  - <s> Sam I am </s>
  - <s> I do not like green eggs and ham </s>
- <s> is a pseudo-word added for convenience
- Compute bigram probabilities:
  - $P(I| < s >), P(am|I), P(</s > |Sam)$
- Solution

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

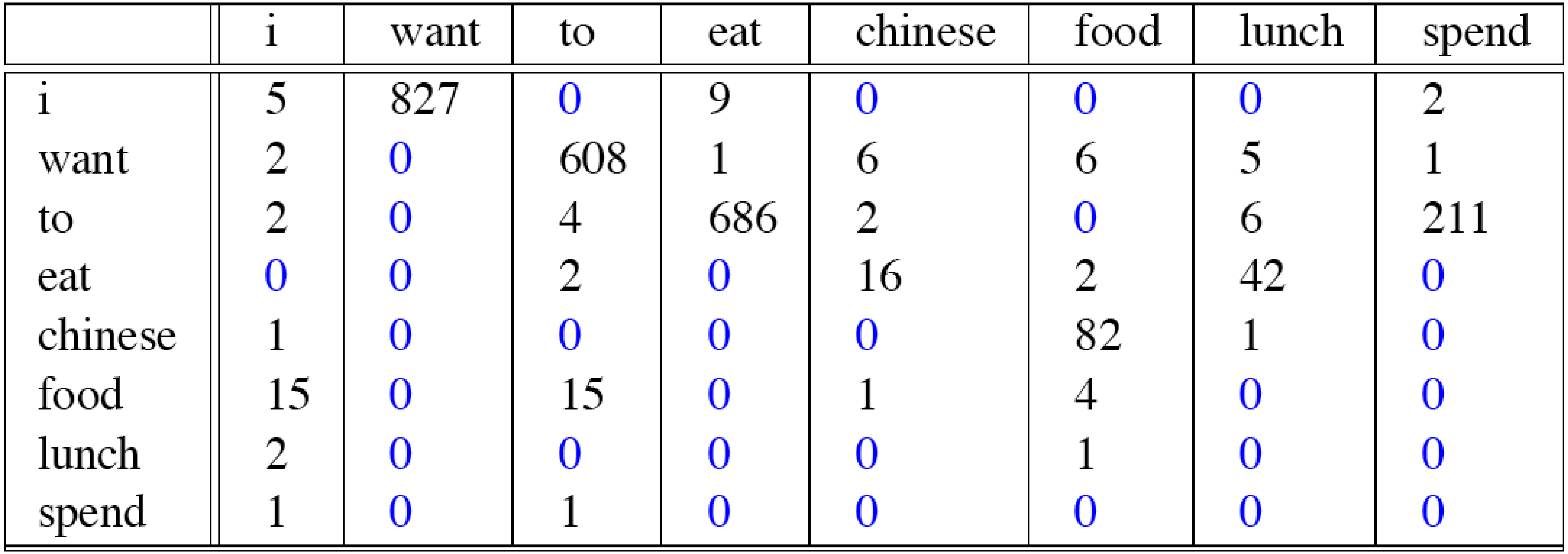
$$P(\texttt{</s>}\,|\,\texttt{Sam}) = \tfrac{1}{2} = 0.5 \qquad P(\texttt{Sam}\,|\,\texttt{am}) = \tfrac{1}{2} = .5 \qquad P(\texttt{do}\,|\,\texttt{I}) = \tfrac{1}{3} = .33$$

LANGUAGE MODEL: BRP CORPUS

- Berkeley Restaurant Project
  - A dialogue system that answered questions about a database of restaurants in Berkeley, California (Jurafsky et al. 1994)
- 9332 Sentences, Vocabulary = 1446
  - can you tell me about any good cantonese restaurants close by
  - mid priced thai food is what i'm looking for
  - tell me about chez panisse
  - can you give me a listing of the kinds of food that are available
  - i'm looking for a good place to eat breakfast
  - when is caffe venezia open during the day
  - …

LANGUAGE MODEL: BRP EXAMPLE

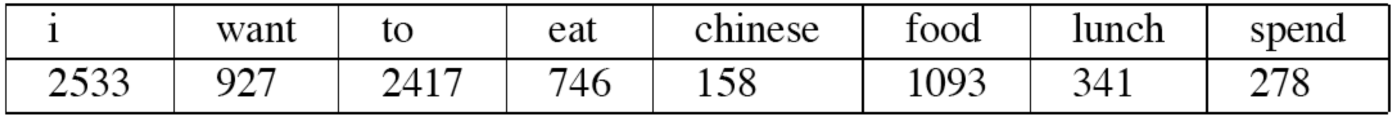
- Bigram frequencies
  - Mostly 0 (Sparse Matrix)



	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

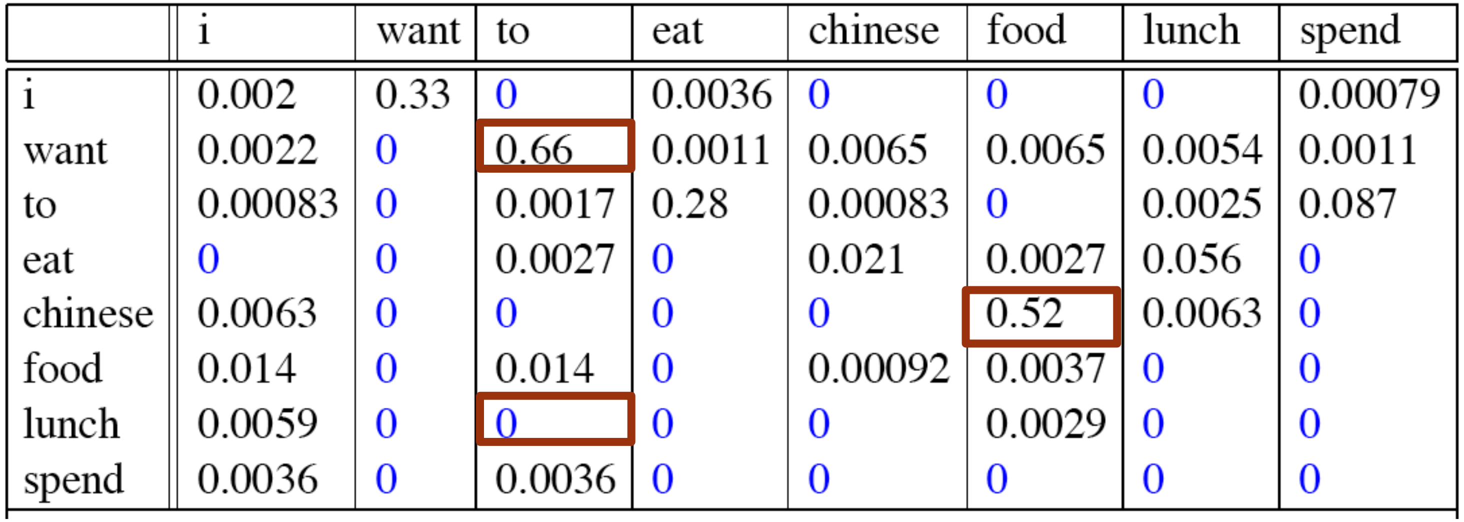
LANGUAGE MODEL: BRP EXAMPLE

- Unigram frequencies



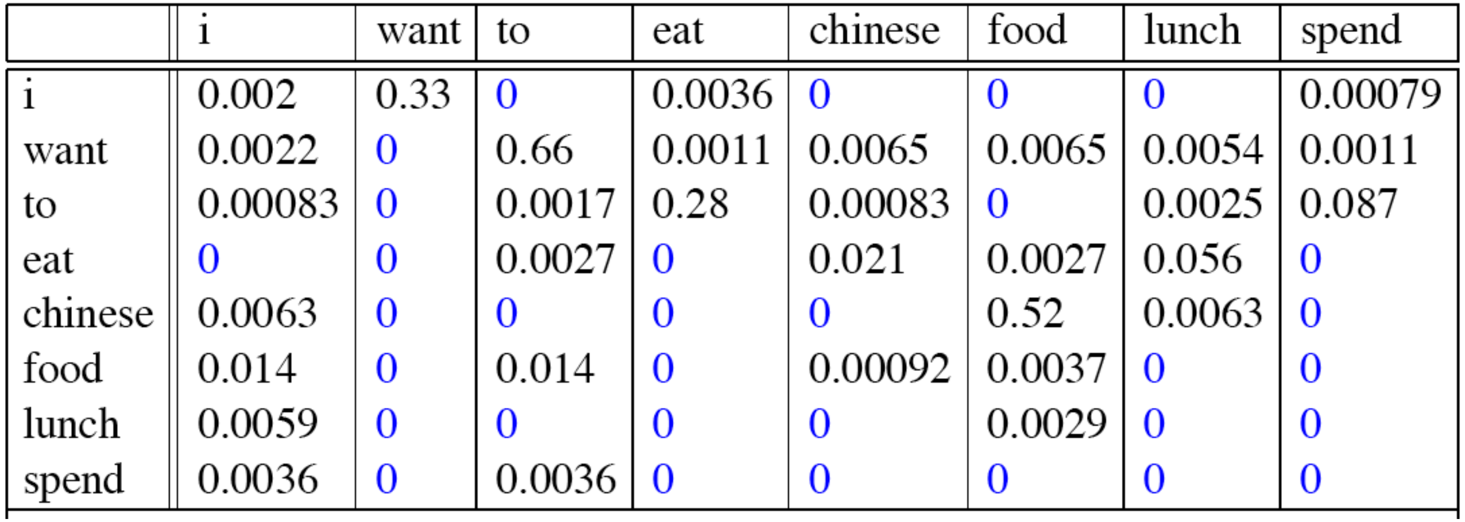
i	want	to	eat	chinese	food	lunch	spend
2533	927	2417	746	158	1093	341	278

- After Normalization



	i	want	to	eat	chinese	food	lunch	spend
i	0.002	0.33	0	0.0036	0	0	0	0.00079
want	0.0022	0	0.66	0.0011	0.0065	0.0065	0.0054	0.0011
to	0.00083	0	0.0017	0.28	0.00083	0	0.0025	0.087
eat	0	0	0.0027	0	0.021	0.0027	0.056	0
chinese	0.0063	0	0	0	0	0.52	0.0063	0
food	0.014	0	0.014	0	0.00092	0.0037	0	0
lunch	0.0059	0	0	0	0	0.0029	0	0
spend	0.0036	0	0.0036	0	0	0	0	0

LANGUAGE MODEL: BRP EXAMPLE



	i	want	to	eat	chinese	food	lunch	spend
i	0.002	0.33	0	0.0036	0	0	0	0.00079
want	0.0022	0	0.66	0.0011	0.0065	0.0065	0.0054	0.0011
to	0.00083	0	0.0017	0.28	0.00083	0	0.0025	0.087
eat	0	0	0.0027	0	0.021	0.0027	0.056	0
chinese	0.0063	0	0	0	0	0.52	0.0063	0
food	0.014	0	0.014	0	0.00092	0.0037	0	0
lunch	0.0059	0	0	0	0	0.0029	0	0
spend	0.0036	0	0.0036	0	0	0	0	0

- P(i|<s>) = 0.25, P(</s>|food) = 0.68
- P(<s> i want chinese food </s>) = ?
- P(<s> i want chinese food </s>) = P(i|<s>) * P(want|i) * P(chinese|want) * P(food|chinese) * P(</s>|food)
- P(<s> i want chinese food </s>) = 0.25 * 0.33 * 0.0065 * 0.52 * 0.68
- P(<s> i want chinese food </s>) = 0.000189618

MODEL EVALUATION

Given

- A statistical model **M**
- A dataset (or corpus) **D**

Objective

- Quantify how good is the model
- Evaluation metric
  - Maximize or minimize

HOW?

Divide data

- **Training** set: To estimate ("learn") the model parameters (e.g. 80%)
- **Test** set: For model evaluation (e.g. 10%)
- **Held-out** (*validation*) set: for hyper-parameter tuning (e.g. 10%)

Hyper-parameter

- Set of model parameters which are not estimated using training set
  - e.g. λ in interpolation (more on this later)

Beware!

- Training on Test set
  - Overfitting! (Cheating)
  - Invalid conclusions

EVALUATION TYPES

Extrinsic

- Model is embedded in an application.
- Measure quality of the model by evaluating the application performance.
- For example: Language model in speech recognition
  - Quality of speech recognition system
- **Computationally Expensive!**

Intrinsic

- Measures quality of model independent of any applications
- For Example: Language model for next word prediction in test set
  - Evaluation metrics designed for intrinsic evaluation
- Comparatively less expensive

LANGUAGE MODEL: EVALUATION

- A good model?
  - Assign higher probability to "real" or "frequently observed" sentences than "ungrammatical" or "rarely observed" sentence
- Extrinsic
  - Put each model in a task
    - spelling corrector, speech recognizer etc.
  - Task dependent metrics
    - How many misspelled words corrected properly
    - How many words translated correctly
- Intrinsic
  - Perplexity

LANGUAGE MODEL: PERPLEXITY

- Best model?
  - Gives highest probability for word sequences (sentences) in the test set
- Perplexity: $PP(W_{test}) = \sqrt[n]{\frac{1}{P(w_1,w_2,w_3,\ldots,w_n)}}$
- Chain rule: $PP(W_{test}) = \sqrt[n]{\frac{1}{\prod_i^n P(w_i|w_1, w_2, \ldots, w_{i-1})}}$
- For bigrams: $PP(W_{test}) = \sqrt[n]{\frac{1}{\prod_i^n P(w_i|w_{i-1})}}$
- **Minimizing perplexity same as maximizing probability**
- Lower perplexity = Better model

LANGUAGE MODEL: PERPLEXITY

- For WSJ Corpus (Training 38 million words, Testing 1.5 million words)
- Why trigrams?

N-gram Order	Unigram	Bigram	Trigram
Perplexity	962	170	109

LANGUAGE MODEL: LANGUAGE GENERATION

- A language model can predict the next word given a sequence of previous words
- Language generation! How?
  - Choose a random bigram (<s>, w) according to its probability
  - Now choose a random bigram (w, x) according to its probability
  - And so on until we choose </s>
  - Then string the words together

<s> I

I want

want to

to eat

eat Chinese

Chinese food

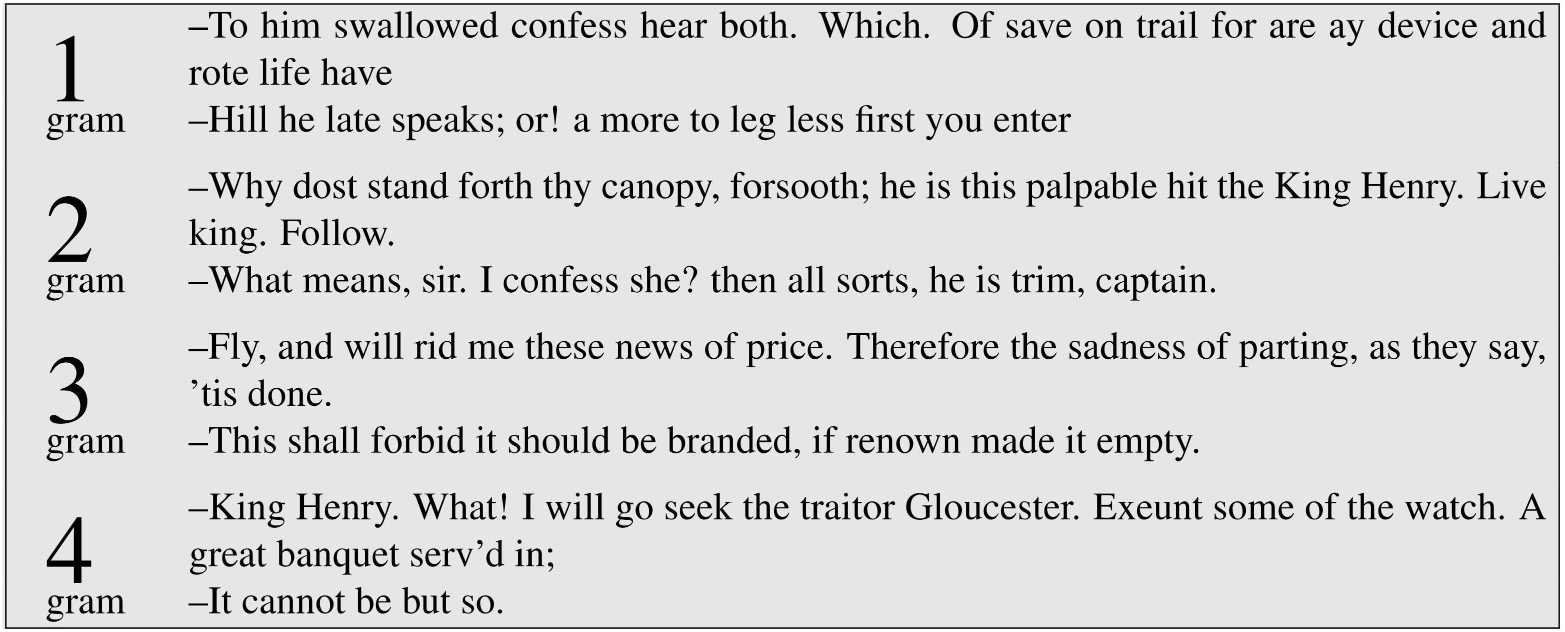
food

</s>

I want to eat Chinese food

LANGUAGE MODEL: SENSITIVITY TO DATA

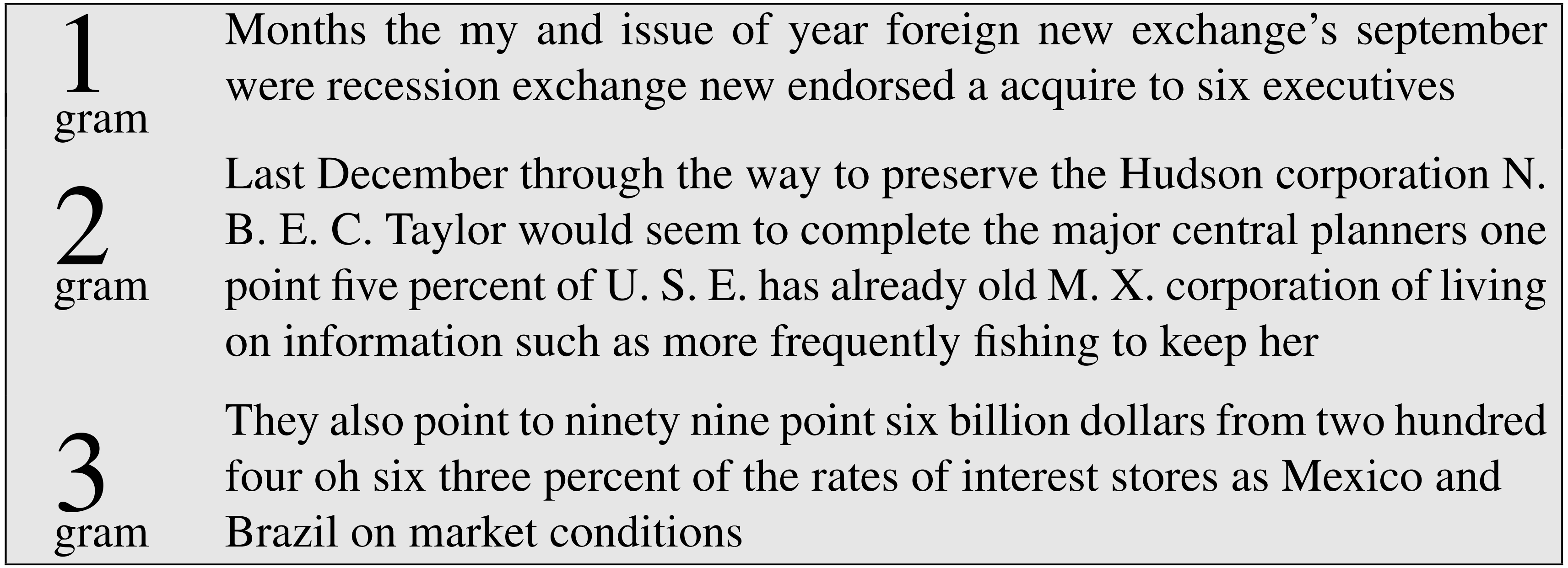
- Language Generation using Shakespeare's work as corpus



1 gram	–To him swallowed confess hear both. Which. Of save on trail for are ay device and rote life have –Hill he late speaks; or! a more to leg less first you enter
2 gram	–Why dost stand forth thy canopy, forsooth; he is this palpable hit the King Henry. Live king. Follow. –What means, sir. I confess she? then all sorts, he is trim, captain.
3 gram	–Fly, and will rid me these news of price. Therefore the sadness of parting, as they say, 'tis done. –This shall forbid it should be branded, if renown made it empty.
4 gram	–King Henry. What! I will go seek the traitor Gloucester. Exeunt some of the watch. A great banquet serv'd in; –It cannot be but so.

LANGUAGE MODEL: SENSITIVITY TO DATA

- Language Generation using Wall Street Journal (WSJ) corpus



1 gram	Months the my and issue of year foreign new exchange's september were recession exchange new endorsed a acquire to six executives
2 gram	Last December through the way to preserve the Hudson corporation N. B. E. C. Taylor would seem to complete the major central planners one point five percent of U. S. E. has already old M. X. corporation of living on information such as more frequently fishing to keep her
3 gram	They also point to ninety nine point six billion dollars from two hundred four oh six three percent of the rates of interest stores as Mexico and Brazil on market conditions

LANGUAGE MODEL: GENERALIZABILITY

- Test and training set can be different!
- Sparsity problem
  - Zero probability bigrams
  - Example
    - P(<s> i want english food </s>) = P(i|<s>) * P(want|i) * P(english|want) * P(food|english) * P(</s>|food)
    - If P(english|want) = 0, the entire sequence has 0 probability
  - Cannot compute perplexity!
- Sparsity increases with the size of N-gram
- Unknown words in test set

LANGUAGE MODEL: SMOOTHING

- We have sparse statistics

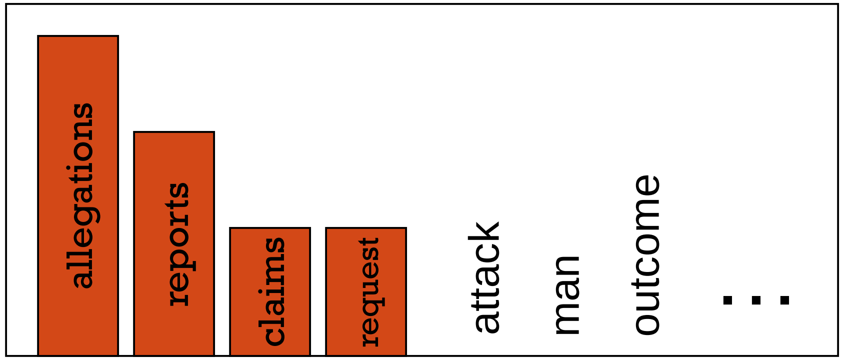
P(w | denied the)
3 allegations
2 reports
1 claims
1 request

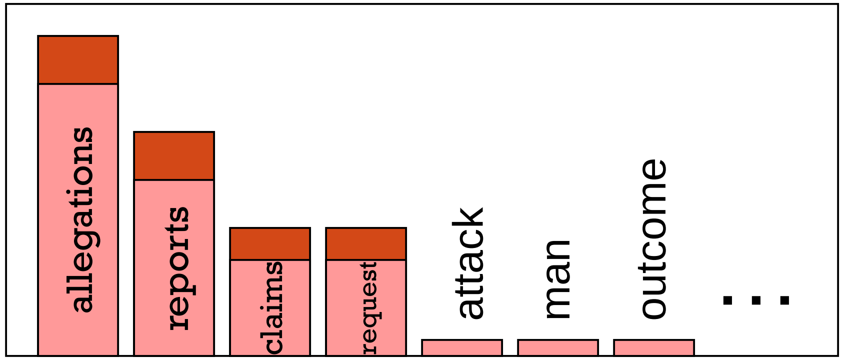
7 total

- Steal probability mass to generalize better

P(w | denied the)
2.5 allegations
1.5 reports
0.5 claims
0.5 request
2 other

7 total



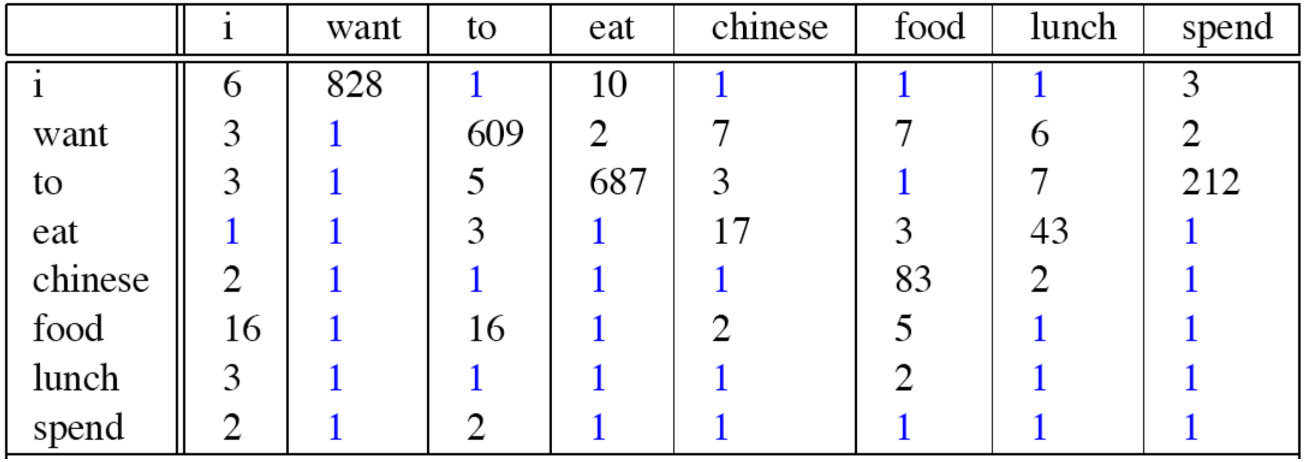


LANGUAGE MODEL: LAPLACE SMOOTHING

- Also known as add 1 smoothing
- Assume that you have seen every N-gram in training corpus one more time than you actually did
  - For unigrams: $P_{Laplace}(w_i) = \frac{c_i+1}{N+V}$
  - For bigrams: $P_{Laplace}(w_n|w_{n-1}) = \frac{C(w_{n-1}\, w_n)+1}{C(w_{n-1})+V}$
- Discounting perspective
  - Adjusted counts (unigrams) $c_i^* = (c_i+1)\frac{N}{N+V}$
  - Discount $d_c = \frac{c^*}{c}$

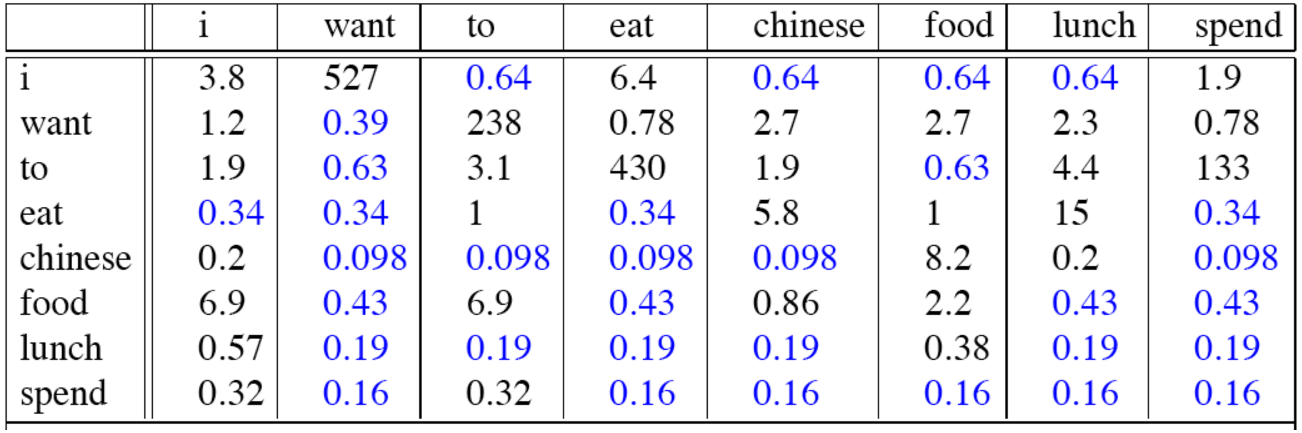
LANGUAGE MODEL: LAPLACE SMOOTHING

- Counts (c)



	i	want	to	eat	chinese	food	lunch	spend
i	6	828	1	10	1	1	1	3
want	3	1	609	2	7	7	6	2
to	3	1	5	687	3	1	7	212
eat	1	1	3	1	17	3	43	1
chinese	2	1	1	1	1	83	2	1
food	16	1	16	1	2	5	1	1
lunch	3	1	1	1	1	2	1	1
spend	2	1	2	1	1	1	1	1

- Adjusted Counts (c*)



	i	want	to	eat	chinese	food	lunch	spend
i	3.8	527	0.64	6.4	0.64	0.64	0.64	1.9
want	1.2	0.39	238	0.78	2.7	2.7	2.3	0.78
to	1.9	0.63	3.1	430	1.9	0.63	4.4	133
eat	0.34	0.34	1	0.34	5.8	1	15	0.34
chinese	0.2	0.098	0.098	0.098	0.098	8.2	0.2	0.098
food	6.9	0.43	6.9	0.43	0.86	2.2	0.43	0.43
lunch	0.57	0.19	0.19	0.19	0.19	0.38	0.19	0.19
spend	0.32	0.16	0.32	0.16	0.16	0.16	0.16	0.16

- Issue?

LANGUAGE MODEL: LAPLACE SMOOTHING

- Issues?
  - Too many unseen N-grams (e.g. bigrams, trigrams etc)
  - Too much probability mass required for unseen N-grams!
- Solution?
  - Add delta smoothing (fraction)
  - Other smoothing methods
- Useful
  - Classification problems
  - Information retrieval (ranking)
  - Domains without sparsity

LANGUAGE MODEL: OTHER SOURCES

- Other sources of information
  - N-gram hierarchy
  - Trigram probability $P(book|opened, the) = 0$
  - Bigram probability $P(book|the) > 0$
- Back-off
  - Use trigram if you have good evidence,
  - Otherwise bigram, otherwise unigram
  - Katz backoff
- Interpolation
  - For trigram language model
    - Weighted average of unigram, bigram and trigram probabilities

LANGUAGE MODEL: INTERPOLATION

- Simple interpolation

$$\begin{aligned}\hat{P}(w_n|w_{n-2}w_{n-1}) &= \lambda_1 P(w_n|w_{n-2}w_{n-1})\\ &+\lambda_2 P(w_n|w_{n-1})\\ &+\lambda_3 P(w_n)\end{aligned}$$

$$\sum_i \lambda_i = 1$$

- Lambdas conditional on context:

$$\begin{aligned}\hat{P}(w_n|w_{n-2}w_{n-1}) &= \lambda_1(w_{n-2}^{n-1})P(w_n|w_{n-2}w_{n-1})\\ &+\lambda_2(w_{n-2}^{n-1})P(w_n|w_{n-1})\\ &+\lambda_3(w_{n-2}^{n-1})P(w_n)\end{aligned}$$

LANGUAGE MODEL: INTERPOLATION

- How to identify correct lambdas?
- lambdas are hyper-parameters
  - Use Held-out set
- Fix the N-gram probabilities (on the training data)
- Then search for lambdas that give largest probability to held-out set:

$$\log P(w_1 \ldots w_n \mid M(\lambda_1 \ldots \lambda_k)) = \sum_i \log P_{M(\lambda_1 \ldots \lambda_k)}(w_i \mid w_{i-1})$$

LANGUAGE MODEL: UNKNOWN WORDS

- Closed vocabulary Task
  - All the words are known
  - No new words in test set
  - **Unrealistic!**
- Open vocabulary Task
  - There might be unknown words in test set
  - More common
- Out of vocabulary words (OOV), represented as a pseudo-word <UNK>
- Training of <UNK> probabilities
  - Create a fixed vocabulary ***V*** of size ***m***
  - During training phase, any word not in ***V*** is changed to <UNK>
  - Treat <UNK> as a normal word
  - While testing, use <UNK> probabilities for any word not in training

LANGUAGE MODEL: NGRAM ISSUES

- Long distance dependencies
  - opened the ______
  - students opened the ______
  - The computer(s) which I had just put into the machine room on the fifth floor is (are) _____
    - crashing
- Storage issues
  - Scales with context size or N
  - More for Interpolation and Backoff
- Synonyms etc.
- Solution?

LANGUAGE MODEL: ADVANCED

- Toolkits
  - Srilm http://www.speech.sri.com/projects/srilm/
  - Kenlm https://kheafield.com/code/kenlm/
- Google N-gram corpus
  - http://ngrams.googlelabs.com/
- Modern day research
  - BERT, XLNet, GPT2 (All neural network based)
  - Word embeddings as a side product
  - https://www.reddit.com/r/SubSimulatorGPT2/comments/dc7lt4/florida_woman_charged_with_murder_says_she_killed/

SUMMARY

- A language model assigns probability to a sequence of words
- It can be used in a variety of tasks including language generation, classification, speech recognition, auto-complete, authorship attribution etc.
- N-gram language model uses Markov assumption to simplify the problem
- An N-gram language model uses MLE to compute N-gram probabilities
- N-gram language models are sensitive to the corpus used
- Smoothing provides better generalization for N-gram models
- Language models are evaluated by dividing the data in trainig, test and held-out set.
- State-of-the-art language models use Neural Networks

QUESTIONS?