

Count-based Language Models

CS 4417B

The University of Western Ontario



John Kelly
@mashedradish

Following



Let's have predictive text write out our conjunctive band names, e.g., Tom Petty and the Heartbreakers.

Write: "[YOUR FIRST AND LAST NAME] and the" and let predictive text complete the rest.

Mine: John Kelly and The Civil War.

9:33 AM - 5 Jan 2018

4 Retweets 39 Likes



71



4



39



Tweet your reply



Mrs Hallahan @heymrshallahan · Jan 8



Replying to @mashedradish

Grainne Hallahan and the kids have been sick. 🤒



1



1



2



John Kelly @mashedradish · Jan 8



I hope they feel better.



Statistical Language Models

[BCC Ch. 1.3.4]

*give prediction based on the probability
learn from previous contexts*

- Attempt to capture probabilities

- Of observing a term or sequence of terms
- Usually given some context

corpus needed

- Captures the sequential structure of a language

- Grammar/word choice

- “There are” more likely than “Their are”

- Idioms

in corpus, this one has higher possibility.

- “Raining cats and dogs” more likely than “Raining dogs and cats”

- Topics

- “peas and carrots” more likely than “peas and briefcases”

use statistics to reflect probability.

Add'l reading: [<https://web.stanford.edu/~jurafsky/slp3/4.pdf>]

Uses

- Predictive text
- Language generation
- Grammar checking
- Evaluating machine-generated text
 - | | | | | | |
|----|----|-----------|------------|------|---------|
| 他 | 向 | 记者 | 介绍了 | 主要 | 内容 |
| He | to | reporters | introduced | main | content |
 - “He, to reporters, introduced the main content.”
 - “He introduced reporters to the main contents.”
 - “He briefed to reporters the main contents.”
 - “He briefed reporters on the main contents.”
- Internals of language models are often useful representations for words and documents

huge amount

of data

required.

Distribution over terms

- Suppose we choose a position uniformly randomly from a corpus.
 - What is the probability that the term at that position is “the”?
 - What is the probability that the term at that position is “coconut”?
- “Term model” provides estimate of $P(t)$.
- If there are m terms in our vocabulary, how many numbers do we need to store to describe this distribution?

$$\sum P(t) = 1$$

little less than m \Rightarrow $m-1$ actually since $P_{\text{max}} = 1 - \sum_{t=0}^{m-1} P(t)$

Conditional Models

- Suppose we choose a position uniformly randomly from a corpus, excluding the last position...
- ...and we're told the word (e.g. "the")
 - What is the probability that the next word is "coconut?"
 - What is the probability that the next word is "the?"
- "(Conditional) bigram model" provides estimate of $P(t_2|t_1)$ *next one's possibility.*
giving the current word .e.g. "coconut", getting
- If there are m terms in our vocabulary, how many numbers do we need to store this distribution?

$$m \times (m-1)$$

\Rightarrow roughly m^2

More complex models

- $P(t_1 | t_2, t_3, t_4)$
- How many parameters? (iClicker)

- A) $m-1$
- B) $m*(m-1)$
- C) $m*m*(m-1)$
- D) $m*m*m*(m-1)$
- E) $m*m*m*(m-1)$

More complex models

- $P(t_1 | t_2, t_3, t_4)$
- How many parameters?
- Suppose a 10000-word vocabulary (modest.)
 - There are 10^{16} 4-grams
 - English Wikipedia has about $10^{9.5}$ words
 - Best-case scenario, 99.9999% of 4-grams never occur

Language Model Sparsity

- Often, we want a language model to *generalize* beyond the corpus it was learned from
 - E.g. for evaluating sentences/text in that language, and to be able to create new text
- Good sparsity: n -grams that don't make sense have probability 0. (“he you bird now”)
- Bad sparsity: Plausible n -grams that happen not to be present in the corpus get probability zero.
- More data reduces bad sparsity.

↑
wrong
case!

Simplifying Assumptions

- Suppose we choose to represent $P(\underline{t_1}, \underline{t_2})$ like so:

↑
given.

- $P(t_1, t_2) = P(t_1) P(t_2)$

- When would this model give exactly the right probability? (What property of language?)

this one is the probability for t_1 and t_2 are independent

- How many parameters? to each other

$$\begin{aligned} P(t_1) &\Rightarrow (m-1) \\ P(t_2) &\Rightarrow (m-1) \end{aligned} \quad \rightarrow 2 \times (m-1).$$

Independence Model

- $P(t_1, t_2, t_3, \dots, t_k) = P(t_1) P(t_2) P(t_3) \dots P(t_k)$
- This is an *independence assumption*.
 - Knowing t_1 tells us nothing about what $P(t_2)$ will be, etc.
- Note under this assumption $P(t_1, t_2) = P(t_2, t_1)$
 - All order information is lost; much like Bag of Words

(Markov) Bigram Model

$$\begin{aligned} & P(< s>, t_1, t_2, \dots, t_k, </s>) \\ &= P(t_1 | < s>) P(t_2 | t_1) \dots P(t_{k-1} | t_k) P(</s> | t_k) \end{aligned}$$

- $< s>$, $</s>$ are special tokens for start and end of sentence
 - This kind of model is only used on whole sentences.
- Probability of next term only depends on term immediately before
- Assigns probabilities to sequences of arbitrary length
- Number of parameters fixed: $m \times (m - 1)$

Trigram Model

$$\begin{aligned} & P(< s>, < s>, t_1, t_2, \dots, t_k, < /s>) \\ &= P(t_1 | < s>, < s>) P(t_2 | t_1, < s>) \dots P(t_k | t_{k-1}, t_{k-2}) P(< /s> | t_k) \end{aligned}$$

- Probability of next term only depends previous two terms
- 4-gram, 5-gram, ... are similar.

Estimating models from data

- $P(t_k | t_1, t_2, t_3, \dots, t_{k-1}) = P(t_1, t_2, t_3, \dots, t_k) / P(t_1, t_2, t_3, \dots, t_{k-1})$
- $P(t_k | t_1, t_2, t_3, \dots, t_{k-1}) \approx C(t_1, t_2, t_3, \dots, t_k) / C(t_1, t_2, t_3, \dots, t_{k-1})$

Smoothing

- Zero probabilities cause problems.
 - In our Markov/bigram/trigram/ngram models, if just one of the probabilities is zero, the whole sequence is given probability zero
- “Smoothing” modifies our probability estimates to avoid zero probabilities.

Laplace Smoothing

- So far, all probability estimates we have seen are derived from counts (of terms, bigrams, trigrams, etc.)
- Laplace smoothing adds 1 to each count before normalizing appropriately.
- Avoids zero probabilities, but doesn't work that well.

Back-off

- If we encounter a never-before-seen 3-gram, maybe we have seen the 2-gram. (Or the term.)
- E.g. never seen “mustard ice cream” but have seen “ice cream”
- “Back-off” methods default to simpler models if the more complicated ones do not have reliable estimates.
 - Note we can’t just “substitute” simpler model or things won’t normalize correctly. “Katz back-off” is one strategy
 - “Stupid back-off” ignores this problem; works well on large corpora. Brants, T., Popat, A. C., Xu, P., Och, F. J., and Dean, J. (2007). Large language models in machine translation. In *EMNLP/CoNLL 2007*.

Context-based Smoothing

- Suppose want $P(T|\text{reading})$ but “reading” was not in our corpus.
- Which of our known words are more likely to appear after an unknown word?
 - “San Francisco”
 - “safety glasses”
 - “drinking glasses”
 - “water glasses”
 - ...
- Kneser-Ney smoothing boosts these “likely continuations”

Applications

- Text generation
- Filtering output from other systems
- Intelligent “spell checking”
- Corpus membership classification

Babbling

- Given a distribution over words, it's possible to draw a word according to that distribution.
 - If $P(\text{coconut} = 0.75)$ and $P(\text{pear} = 0.25)$, “coconut” will appear about 3 times out of 4, and pear about 1 out of 4.
- “Babbling” is drawing a sequence of words, always conditioning on the most recently generated word(s) to produce the next one.
- Gives a sense of what structure is captured by the model.

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

Figure 4.3 Eight sentences randomly generated from four *N*-grams computed from Shakespeare's works. All characters were mapped to lower-case and punctuation marks were treated as words. Output is hand-corrected for capitalization to improve readability.

Filtering

- Statistical machine translation methods often produce several plausible translations
 - We can get the translation model to “babble” likely sentences
 - Choose the one with the highest probability under the language model
- Speech-to-text also produces statistical models over possible sentences
- Language modeling allows both of these to be improved with huge amounts of “unlabeled” data

Intelligent “spell checking”

Class will end in five minuets.

Corpus membership classification

- “Generative” model for classification.
- E.g., classify passage as Stephen King or Shakespeare:
 - Build language model for Stephen King
 - Build language model for Willy Shakes
 - Ask probability of passage belonging to each, see which one gives higher probability.

Summary

- Language models that predict next word given past context – (sometimes called “causal” language models)
- Challenge to naïve approaches: Data sparsity
- Applications
 - Generation – create new text
 - Evaluation – assess whether text is plausible
 - Classification – assess relative likelihood of authorship
 - Many more

End of the session