DATA.STAT.840 Statistical Methods for Text Data Analysis

Exercises for Lecture 5: N-grams

Exercise 5.1: Bigram probabilities.

(a) Are the following probabilities possible in an bigram model? $p(w_1='\text{rock}')=0.01$, $p(w_2='\text{band}')=0.003$, $p(w_2='\text{band}'|w_1='\text{rock}')=0.4$. Prove why/why not. Derive an inequality between $p(w_1)$, $p(w_2)$ and $p(w_2|w_1)$ for what probabilities are possible. Hint: consider the Bayes rule.

(b) Consider the sentence "The whole of science is nothing more than a refinement of everyday thinking." (Albert Einstein, *Physics and Reality*, 1936). Compute the probability of the sentence in a bigram model using the following unigram probabilities: $p('\text{the}')=0.03$, $p('\text{whole}')=0.0001$, $p('\text{of}')=0.01$, $p('\text{science}')=0.0003$, $p('\text{is}')=0.02$, $p('\text{nothing}')=0.0002$, $p('\text{more}')=0.001$, $p('\text{than}')=0.0009$, $p('\text{a}')=0.025$, $p('\text{refinement}')=2\cdot 10^{-6}$, $p('\text{everyday}')=6\cdot 10^{-6}$, $p('\text{thinking}')=3\cdot 10^{-5}$. You need to choose some corresponding bigram probabilities so that they satisfy the condition you derived in (a).

Report your proof and computations.

Exercise 5.2: Theoretical n-gram properties.

(a) Suppose you need to generate a document of length M words. Show that if the n in an n-gram model is at least as large as M, the n-gram model can represent all statistical dependencies that might exist in the language needed to generate the document. So that, for example, a 5-gram model can represent all dependencies needed to generate sentences of 5 words.

(b) Consider a simplified version of the maximum a posteriori estimation of n-gram probabilities described on the lecture. Suppose all pseudocounts in the Dirichlet priors use the same shared value, $\alpha_{v|[w_1,\ldots,w_{n-1}]}=\alpha_{shared}$ for all vocabulary terms v and all contexts $[w_1,\ldots,w_{n-1}]$ where α_{shared} is the shared value. This results in estimates that are simple smoothed proportional counts. This kind of smoothing is called **Laplace smoothing** when $\alpha_{shared}=1$ and **Lidstone smoothing** otherwise.

- ○ Show that in this setting, the maximum a posteriori estimate (as shown on the course slides) for a n-gram probability can be written as a weighted average of two terms: (1) the maximum likelihood estimate of the probability and (2) a uniform distribution over the vocabulary.
- ○ Show that the mixing weight in the weighted average depends on the number of occurrences of a n-gram context compared to $V\alpha_{shared}$ where V is the vocabulary size.
- ○ For an individual n-gram context, how should α_{shared} be chosen so that the weight of the data is greater than the weight of the prior?

Report your proofs.

(exercises continue on the next page)

Exercise 5.3: More adventures of Robin Hood, and a new journey to Mars.

(a) Download the following ebooks from Project Gutenberg: Howard Pyle's "The Merry Adventures of Robin Hood", and Stanley G. Weinbaum's 1934 science fiction story "A Martian Odyssey". Process them separately: tokenize, turn to lowercase, and find a vocabulary. No need to lemmatize the words or prune the vocabulary.
(For more about these works see https://en.wikipedia.org/wiki/The_Merry_Adventures_of_Robin_Hood and https://en.wikipedia.org/wiki/A_Martian_Odyssey.)

(b) For both books, train n-gram models with the following maximum values of n: 1, 2, 3, 5.

(c) For each books, using each trained n-gram model, generate 2 new paragraphs of text. Discuss the results and the difference between the different values of n. Do the results with large n show memorization (can you find the generated paragraphs, or long parts of them, in the text of the book)?

(d) For each book, generate a paragraph of text starting with "The moon", using n=2, 3, and 5. Can you easily tell which book the generated text is likelier to belong to?

Report the requested texts and your code.

Exercise 5.4: N-gram abilities versus AI chatbot abilities

(a) Can n-grams learn to write text from a corpus where the order of letters in words has been reversed? (For example, "Ti saw eth tseb fo semit, ti saw eth tsrow fo semit.") Why/why not? If they can, what limitations are there in what can be learned?

(b) Can n-grams learn to write text from a corpus where each letter has been rotated on step backwards in the alphabet (For example "Hs vzr sgd adrs pe shldr, Hs vzr sgd vnqrs pe shldr.") Why/why not? If they can, what limitations are there in what can be learned?

(c) Can n-grams learn to write text from a corpus where the order of the intermediate letters has been randomly scrambled? (For example, "Dolaonwd the flionwlog ekobos form Pjeocrt Gteubrneg".) Why/why not? If they can, what limitations are there in what can be learned?

(d) Can n-grams learn the general rule that an opening parenthesis must be followed by a closing parenthesis? Why/why not? If they can, what limitations are there in what can be learned?

(e) Prove that n-gram models can be used to predict a missing middle word in a string like "the cat w into the box" where w is a missing word, and "the cat"=Left and "into the box"=Right are its contexts to the left and right. To do so, prove that the probability p(w|Left,Right) can be rewritten using only n-gram style probabilities that give the probability of a word given its n-1 previous words. Hint: use the Bayes rule and the chain rule of probabilities.

(f) Optional extra part: ask a modern deep generative chatbot such as ChatGPT or https://deepai.org/chat (or any local offline chatbot), to continue the example sentences "Ti saw eth tseb fo semit, ti saw eth tsrow fo semit.", "Hs vzr sgd adrs pe shldr, Hs vzr sgd vnqrs pe shldr", and "Dolaonwd the flionwlog ekobos form Pjeocrt Gteubrneg". Discuss the results.