Dmitrii Obideiko
Suraj Janakiraman
10-02-22
Course: CS 4395.001
Instructor: Dr.Karen Mazidi.

Ngrams Narrative

a) what are n-grams and how are they used to build a language model

An n-gram refers to the range of text from one word to the other word inside a window of text. For example, in the previous sentence, (the range) is an n-gram where the range of words in the window of text between the and range is 2. In this case, (the range) is a bigram. This means that the range of words inside the text (the range) is 2. N-grams ranging beyond 3 words are simply called n-grams where n refers to the range of words from one word to another inside a window of text. N-grams are utilized to build a probabilistic language model which is highly dependent on the corpus of text being used.

b) list a few applications where n-grams could be used

Examples of applications where n-grams could be used include but are not limited to: "spelling error detection and correction, query expansion, information retrieval with serial, inverted and signature files, dictionary look-up, text compression, and language identification." (Robertson et. Willett, 48).

c) a description of how probabilities are calculated for unigrams and bigrams

For unigrams, the probability of a specific unigram or set of unigrams occurring inside of a sample text (a test sentence or test file for example) is calculated by first comparing the sample text with the unigram dictionary of words in the corpus. There are several probabilities that can be calculated: Good Turing probability, Laplace Probability, and Log Probability.
With bigrams however, the probabilities are calculated by checking whether a key word or unigram/token is inside of the first token of the bigram, as well as the second token of the bigram.
The good turning probability is calculated based on the following:

d) the importance of the source text in building a language model

Source text is extremely important in building a language model. The source text contains the corpus for building the probabilistic language model. Depending on what source text is being used can greatly influence the results in the language model.

e) the importance of smoothing, and describe a simple approach to smoothing

The sparsity problem of having zero probability in terms of computing the probability that a test object (test sentence/test file) is inside the corpus or source text is an issue. Laplace smoothing fills in the probability values of 0 with a value a little higher than 0 as a method for solving the sparsity problem.

f) describe how language models can be used for text generation, and the limitations of this approach

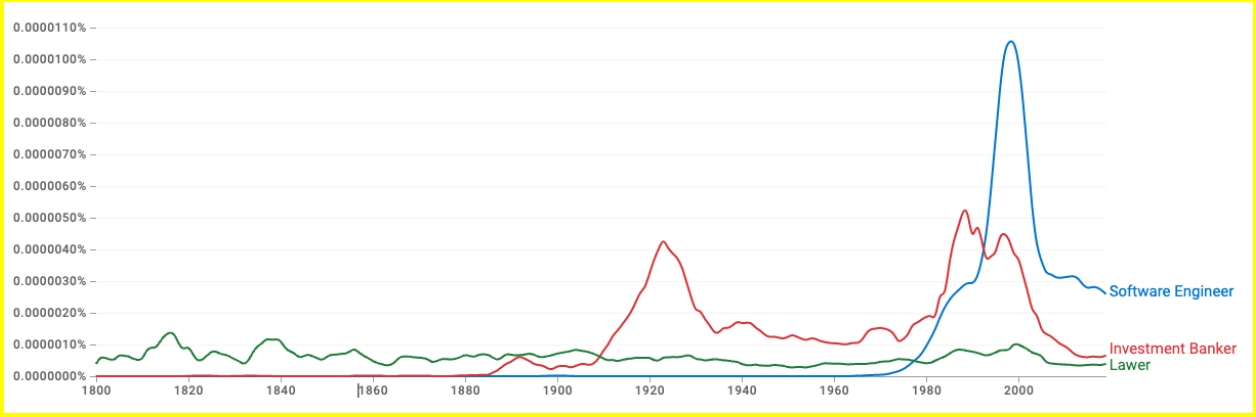
There are two methods used for text generation: the naive approach, and the NLTK generator approach. The naive approach is a function that looks through all the bigram probabilities and checks for the bigram with the start word in the first position that has the highest probability. This method only works for small sample sizes of text. ‘
The second approach is to use the NLTK generator which can generate better results from a larger corpus/source text. The NLTK generator can be called using the nltk.generate() function.

g) describe how language models can be evaluated

For any NLP application, including language models there are two methods of evaluation: intrinsic and extrinsic evaluations. Extrinsic evaluations require humans to evaluate the result of the language model using predefined metrics. Extrinsic evaluations are time-consuming and costly. Intrinsic evaluations use metrics, not necessarily predefined, to compare language models. One such metric is called perplexity. Perplexity measures how well a language model predicts the text inside of the test data. The sample size of the test data is small when calculating perplexity.

h) give a quick introduction to Google’s n-gram viewer and show an example

Google’s n-gram viewer is an online search engine that charts the frequencies of a set of strings by using a yearly count of n-gram. The example you can see below compares the trends of the following terms: Software Engineer, Investment Banker, and Lawyer.



References:

Robertson, Alexander M, and Peter Willett. “Applications of n-Grams in Textual Information Systems.” Journal of documentation 54.1 (1998): 48–67. Web.