Alan Perez

CS4395.001

N-Gram Overview

An N-gram model is a type of probabilistic language model that is used to predict the sequence of words. The N-grams are essentially used to build the foundation of a language model and the n-gram can be seen as a sliding window over text, n words at a time are the sequence of “n” words, while a unigram takes one word, bigram takes two words, and a trigram takes 3. The choice of the corpus (body of text) influences the language model so the richer the body then the more accurate the model would be compared to a body of text that’s sparse. There are various applications that an n-gram could be used for from developing language models to calculating probability. A few examples of an n-grams application can be used as spelling error, word prediction, autocomplete, etc. The way the probability of a unigram is calculated is by P(w1) = count of w1 / number of tokens in text. Bigrams are calculated by the frequency of the previous word.

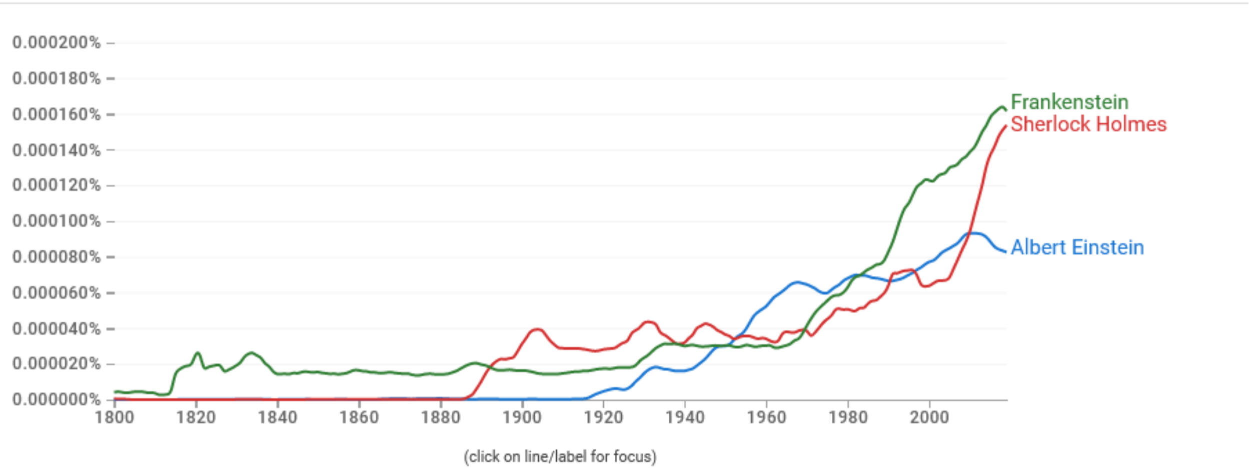
The source text is extremely important since it requires data to build a language model. When a language model has a rich body of text then it will greatly influence the language model, if it has a bad body of text then the results would be bad. The richer the text, then the more accurate the probability of the language model will be.

Smoothing addresses the issue caused by sparsity and adjusts the probabilities in a model, allowing the distribution to be smoother. The simple approach is using Laplace which is to add 1 to the zeros so that it’s not zero.

The way language models are used for text generation is by creating probability dictionaries which are converted to probabilities. The limitation of this is that due to it being extremely dependent on the data being used there’s a possibility of it being inaccurate due to the body of text being used.

The ways language model can be evaluated in an Extrinsic way which has human annotators evaluate the result using predefined metrics. The only downside to this evaluation approach is that it’s time-consuming and expensive since it’s being done by humans. The other method is Intrinsic evaluation which uses models to compare, and it’s done with small test data

Google’s NGram displays user-selected words or phrases in a graph and shows the frequency. It’s entirely made up of scanned books that are available in Google books, the graph itself is labeled with an X-axis that shows the year of the books, and a Y axis shows the frequency of the ngrams


Frankenstein
Sherlock Holmes
Albert Einstein
(click on line/label for focus)