An N-gram is a sliding window of size N over a text. For example, a unigram will of the text "The car is red" will be [the, car, is, red]. With these n-grams, one is able to create a probabilistic model of language, however, the choice of corpus greatly influences the language model. The probabilities in an n-gram language model can be viewed as a maximum likelihood estimate. This means that we can use smoothing, like the LaPlace smoothing or Modified Good-Turing smoothing, to compute the probabilities.

Some applications where n-grams can be used are spelling error detection and correction, query expansion, and language identification.

Unigram probabilities are calculated by taking the count of the frequency a word occurs and dividing it by the total number of words. Bigram probabilities are calculated by dividing the number of times a bigram appears in the given corpus by the total number of times the first word in the given bigram appears in the same corpus.

The source text in building a language model is very important as it will affect the outcome of the probabilities. The smaller the source text, the less accurate the probabilities will be. In addition, generally higher n-grams will work better than lower n-grams.

Smoothing is a solution to the sparsity problem where not every possible n-gram will be in a dict or list. Smoothing works by filling in 0 values with a bit of probability mass. A simple approach to this is using LaPlace smoothing, where one is added to the bigram count and the total vocabulary size is added to the unigram count of the first word in the bigram.

Language models can be used for text generation by creating a probability distribution over a given corpus to predict the most likely next word in a sentence. The limitations of this is that the computed probability distribution is largely dependent on the given corpus.

Language models can be evaluated by the perplexity. Perplexity is the inverse probability of seeing the words we observe, normalized by the number of words. A low perplexity is ideal.

Google's n-gram viewer is a search engine that charts word frequencies from a large corpus of books. It can be used to show cultural and language changes as it is reflected in the corpus of books. For example, searching up 'Albert Einstein' will show the frequency of the words in the corpus of books. Google's n-gram viewer will show a rise of the frequency of the words around 1920 and a much larger increase past the 1940s. This can be presumably because Albert Einstein's theories were not largely looked at until those times.