Hamna B Mustafa

hbm170002

CS 4395 Assignment 5

**Ngrams**

1. **what are n-grams and how are they used to build a language model**

N-grams are sliding windows over text of length n. They are used to learn about text from sequences of words. A unigram takes one word at a time, a bigram takes 2 words at a time, and so on. Thus, an ngram takes n words at a time. N-grams can be used to create a probabilistic model of language. Given enough training data – a large corpus – ngrams can be used to split the data and count the number of times an ngram appeared in the data. This information can be used to calculate the probability of some data occurring. Ngrams can also generate text by using the probabilities of each ngram.

1. **list a few applications where n-grams could be used**

Ngrams can be used in many different applications. They can be used in spell-check applications, machine translation, speech recognition as well as auto search or auto suggestions.

1. **a description of how probabilities are calculated for unigrams and bigrams**

In order to calculate probabilities for unigrams and bigrams, we first find the count of each unigram and bigram in the corpus. Then, the probability of a unigram is the count of that unigram divided by the total number of unigrams. The probability of a bigram is *P*(*w*1,*w*2) = *P*(*w*1)*P*(*w*2|*w*1) , where *P*(*w*1) is the probability of the first word in that bigram (which is a unigram) and *P*(*w*2|*w*1) is the count of the bigram divided by the count of the first word in the bigram (count of *w*1). We can use different smoothing methods such as laplace smoothing or Good-Turing smoothing.

1. **the importance of the source text in building a language model**

The source text is very important when using ngrams to build a language model because the model is going to depend heavily on the source text. For example, a model built based on a science textbook will have very different ngrams as compared to a model built on a Shakespeare text. The science textbook model would generate more scientific sentences and have higher probabilities of scientific words as compared to the Shakespeare model which would generate Shakespearean text.

1. **the importance of smoothing, and describe a simple approach to smoothing**

Smoothing is important due to the sparsity problem. Because the corpus is not going to contain every ngram possible, the probability of a test ngram that doesn’t exist in the corpus will then become 0. In order to prevent the probabilities from zeroing out, we can use smoothing. Smoothing is the practice of filling in zero values with a little bit of probability of the overall mass to make for a smoother distribution. One technique is called Laplace smoothing. In Laplace smoothing, 1 is added to all counts in order to prevent any of them being 0. Then, the total vocabulary size is added to the denominator to offset the 1 and balance it out.

1. **describe how language models can be used for text generation, and the limitations of this approach**

Language models can be used for text generation by converting the ngrams to probabilities. Then probability dictionaries are created. One naïve approach to generating language is then looking through the ngram probability dictionaries and finding an ngram with a start word in the first position that has the highest probability. It then adds this to the phrase being generated. It continues to do this until some end token is added, such as a period. The limitations to this approach are that for the generation to have good results, a large corpus is needed and creating dictionaries of counts over such large corpuses can be very time consuming. Further, the higher ngrams are also needed for better results.

1. **describe how language models can be evaluated**

NLP Language models can be evaluated using extrinsic and intrinsic evaluation. Extrinsic evaluation is having human annotators evaluate the results using predefined metrics. Intrinsic evaluation is using some metrics, such as perplexity, to compare models. We can set aside some test data and compare our model’s predictions to the test data to evaluate its accuracy.

1. **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 displays phrases selected by users in a graph that shows how they have occurred in Google Ngram viewer’s corpus. Google Ngram viewer’s corpus is made up of the scanned books available in Google Books. The following photo is an example of the ngram viewer:

Chart, line chart

Description automatically generated