Prof. Dr. Andreas Hotho,
M.Sc. Janna Omeliyanenko
Lecture Chair X for Data Science, Universität Würzburg

9. Exercise for "Sprachverarbeitung und Text Mining"

21.01.2022

1 Knowledge Questions

1. Describe the idea of an N-gram language model in your own words.

> An N-gram language model is an approach to learn a word distribution (N-gram distribution) that best reflects a training corpus (maximum likelihood). A language model trained in this way estimates the next word in a sequence over a probability distribution, given N-1 antecedent words.

2. Give 3 possible applications of a language model in the field of NLP.

> - Automatic Speech recognition
> - Language Detection
> - Spelling correction
> - Handwriting and character recognition
> - Machine translation

3. The Brown corpus has long been used as a representative corpus for the English language. The Google Crawl presented in the lecture is much larger, and contains about 13,500,000 word forms, as opposed to about 300,000 in the Brown corpus. Thus, it also contains many more word forms than English dictionaries. Explain how this phenomenon can occur.

- Numbers
- Names
- Spelling mistake
- Acronyms

4. In the lecture, Shannon's method for generating text was introduced. Describe how resulting sentences change when using more and more complex language models (i.e. first unigrams, then bigrams, etc...).
   Are more complex language models always better? What role does the training corpus play in this?

   The sentences make more and more sense the larger the N-grams used. However, if N is too large, N-grams become so specific that they occur only uniquely in the corpus. Thus, the training corpus is purely memorized. Therefore, Shannon's method requires choosing a sufficiently small N-gram size to allow generalization over the used training corpus.

5. Explain the concepts of extrinsic evaluation and intrinsic evaluation based on the example of language models. How do both concepts differ?

   Extrinsic (externally stimulated) evaluation means the integration of a language model into another application (e.g. machine translation,...) and the evaluation of this system. Intrinsic evaluation, on the other hand, tries to evaluate the quality of the language model directly. An extrinsic evaluation is much more complex, but has the advantage that the benefits of the model can be shown directly. An intrinsic evaluation method such as Perplexity reduction, for example, shows how well a language model is capable of modeling the underlying data, but does not guarantee that this "better language model" will actually produce superior results in an application.

6. Justify why it is unlikely that there will ever be a corpus large enough to contain reliable counts of rare N-grams. How can language models deal with very rarely occurring or even unknown words? What possible solutions were shown in the lecture?

   With language being actively used by humans to communicate, new words, spelling adaptations, and spelling errors continuously develop and result in novel and rare N-grams. This is compounded by a large number of words

occurring very rarely (according to Zipf's distribution). N-grams that include these words are thus even rarer. Approaches to deal with unknown and rarely occuring N-grams include:

- The introduction of special <UNK> tokens for unknown words (gained e.g. through separate training- and development-sets)
- Smoothing (Laplace Smoothing, Good-Turing Smoothing, Kneser-Ney Smoothing, Witten-Bell Smoothing,...)

2 Smoothing

Given are the following sentences:

```
beer is a difficult lecture
to brew beer is difficult
i brew beer in the lecture
in the lecture i drink beer
the lecture is difficult
```

Note: In addition to the words in the sentence, consider the sentence-start token <s>. A sentence ending token should not be used in this task.

a) First, create an overview table with the counts of all bigrams that occur.

	<s>	in	to	beer	lecture	is	a	the	difficult	drink	brew	I
<s>		1	1	1				1				1
in								2				
to											1	
beer		1				2						
lecture						1						1
is							1		2			
a									1			
the					3							
difficult					1							
drink				1								
brew				2								
I										1	1	

b) Using the created counts, calculate the probability of the following sentence each with unigrams and bigrams (calculate unigrams without sentence-start tokens):
`lecture is a difficult beer`

$$
\begin{aligned}
P_{bi}(s) &= P(lecture| < s >) \cdot P(is|lecture) \cdot P(a|is) \cdot P(difficult|a) \cdot P(beer|difficult) \\
&= 0 \\
P_{uni}(s) &= \frac{4}{26} \cdot \frac{3}{26} \cdot \frac{1}{26} \cdot \frac{3}{26} \cdot \frac{4}{26} \\
&= \frac{144}{26^5}
\end{aligned}
$$

c) Calculate the perplexity for the sentence given above, for bigrams and unigrams.

$$
\begin{aligned}
PP_{bi}(s) &= \sqrt[5]{\frac{1}{P(lecture| < s >) \cdot P(is|lecture) \cdot P(a|is) \cdot P(difficult|a) \cdot P(beer|difficult)}} \\
&= \infty \\
PP_{uni}(s) &= \sqrt[5]{\frac{1}{\frac{4}{26} \cdot \frac{3}{26} \cdot \frac{1}{26} \cdot \frac{3}{26} \cdot \frac{4}{26}}} \\
&= \frac{26}{\sqrt[5]{144}} \approx 9.623
\end{aligned}
$$

d) Smooth the bigram counts with Laplace-smoothing and calculate the probability and perplexity of the above sentence again. It is sufficient to smooth only the bigrams that are used in the sentence.

$$
\begin{aligned}
P^*(lecture| < s >) &= \frac{1}{5+12} &&= \frac{1}{17} \\
P^*(is|lecture) &= \frac{1+1}{4+12} &&= \frac{2}{16} \\
P^*(a|is) &= \frac{1+1}{3+12} &&= \frac{2}{15} \\
P^*(difficult|a) &= \frac{1+1}{1+12} &&= \frac{2}{13} \\
P^*(beer|difficult) &= \frac{1}{3+12} &&= \frac{1}{15}
\end{aligned}
$$

$$
\begin{aligned}
P(s) &= P^*(lecture| < s >) \cdot P^*(is|lecture) \cdot P^*(a|is) \cdot P^*(difficult|a) \cdot P^*(beer|difficult) \\
&= \frac{1}{17} \cdot \frac{2}{16} \cdot \frac{2}{15} \cdot \frac{2}{13} \cdot \frac{1}{15} \\
&= \frac{8}{17 \cdot 16 \cdot 15^2 \cdot 13}
\end{aligned}
$$

$$
PP(s) = \sqrt[5]{\frac{17 \cdot 16 \cdot 15^2 \cdot 13}{8}} \approx 9.99
$$

e) Smooth the bigrams with Good-Turing-smoothing and calculate the probability and perplexity of the sentence again. It is sufficient to smooth only the bigrams that are used in the sentence.

$$
c^*_{GT}(\text{unseen}_{bigram}) := \frac{N_1}{\underbrace{N_0}_{\substack{\#\text{ unseen but within the} \\ \text{vocabulary of possible bigrams}}}}
$$

$$
\begin{aligned}
c^*_{GT}(X_c) &= (c+1) \cdot \frac{N_{c+1}}{N_c} \\
N_3 &= 1 \\
N_2 &= 4 \\
N_1 &= 15 \\
N_0 &= 144 - 15 - 4 - 1 = 124 \\
c^*(X_0) &= 1 \cdot \frac{15}{124} = 0.121 \\
c^*(X_1) &= (1+1)\frac{4}{15} = 0.533 \\
c^*(X_2) &= (2+1)\frac{1}{4} = 0.75
\end{aligned}
$$

This results in the new bigram probabilities:

$$
\begin{aligned}
P^*(w_j|w_i) &= \frac{c^*_{GT}(w_i, w_j)}{\sum_x c^*_{GT}(w_i, w_x)} \\
P^*(lecture| < s >) &= \frac{0.121}{7 \cdot 0.121 + 5 \cdot 0.533} = \frac{0.121}{3.513} \\
P^*(is|lecture) &= \frac{0.533}{2.276} \\
P^*(a|is) &= \frac{0.533}{2.493} \\
P^*(difficult|a) &= \frac{0.533}{1.863} \\
P^*(beer|difficult) &= \frac{0.127}{1.864} \\
P(s) &= 0.00003204 \\
PP(s) &= \sqrt[5]{P(s)^{-1}} \approx 7.922
\end{aligned}
$$

3 Shannons Method for Text Generation

In the lecture, Shannon's method for text generation was presented using Shakespeare as an example.
Download the file
`Lewis_Carroll_Alices_Adventures_in_Wonderland.txt`
from WueCampus and determine the unigram, bigram and trigram counts of this novel.
Generate one sentence each for unigrams, bigrams, and trigrams using Shannon's method.
Begin with:

- <s>
- She
- She had

Always choose the N-gram with the highest probability.
Note: Treat punctuation marks as separate tokens.

`http://guidetodatamining.com/ngramAnalyzer/`

> Trigramm:
> `She had never been in a low voice , to see if ...`
>
> Bigramm:
> `She had been changed for the King , they were...`
>
> Unigramm:
> `, , , , , , ...`