Methods in Computational Linguistics
Master of Science *Computational Linguistics*
Exercise: Language Models

apl. Prof. Dr. Sabine Schulte im Walde
Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart

January 14, 2020

n-Grams and Language Models

You are provided a play corpus *wiki-en-flower.txt* extracted from an English *Wikipedia* corpus. You will use this corpus to work with n-grams.

1. Tokenise the corpus:

   ```
   cat wiki-en-flower.txt | tr ’ ’ ’\n’ > wiki-en-flower_token.txt
   ```

2. Determine the number of word tokens and the number of word types in the corpus.

   Hint: Use the Unix commands `sort, uniq` and `wc`.

3. Generate the bigrams and the trigrams that appear in the corpus.

   Hint: Use the Unix commands `tail` and `paste`.

4. How many bigram and trigram types and tokens does the corpus have?

5. Name two bigrams and two trigrams that contain the word *sunflower* and appear more often than once in the corpus. How often do these bigrams and trigrams appear in the corpus?

6. Estimate the probability of the bigram *sunflower seeds* using *maximum likelihood estimation*.

7. Calculate the probability of the sentence *Manitoba is the largest producer of sunflower seeds* using the bigram probabilities.

Smoothing

1. Determine the unigram frequencies for the four word forms *and, of, sunflower, seeds*, and the bigram frequencies for the 16 bigram combinations of these four word forms.

2. Calculate the bigram probabilities for the 16 bigram combinations.

3. Apply *Laplace* smoothing to the bigram frequencies and the bigram probabilities.

4. Compare the following two language models using perplexity on the basis of bigrams. The test set contains only one sentence: *That is complete nonsense!*

   Assume that the bigram probability that a sentence starts with *That* is 1.

   Model 1:

	That	*is*	*complete*	*nonsense*	*!*
That	0.00	0.28	0.13	0.11	0.10
is	0.00	0.00	0.22	0.30	0.01
complete	0.00	0.02	0.03	0.33	0.03
nonsense	0.00	0.00	0.09	0.11	0.41
!	0.40	0.00	0.00	0.00	0.00

   Model 2:

	That	*is*	*complete*	*nonsense*	*!*
That	0.00	0.22	0.19	0.14	0.10
is	0.00	0.00	0.12	0.20	0.02
complete	0.00	0.05	0.05	0.21	0.01
nonsense	0.00	0.00	0.15	0.18	0.41
!	0.35	0.00	0.00	0.00	0.00

5. Explain why an improved language model within a statistical machine translation system might improve the overall quality of the automatic translations.