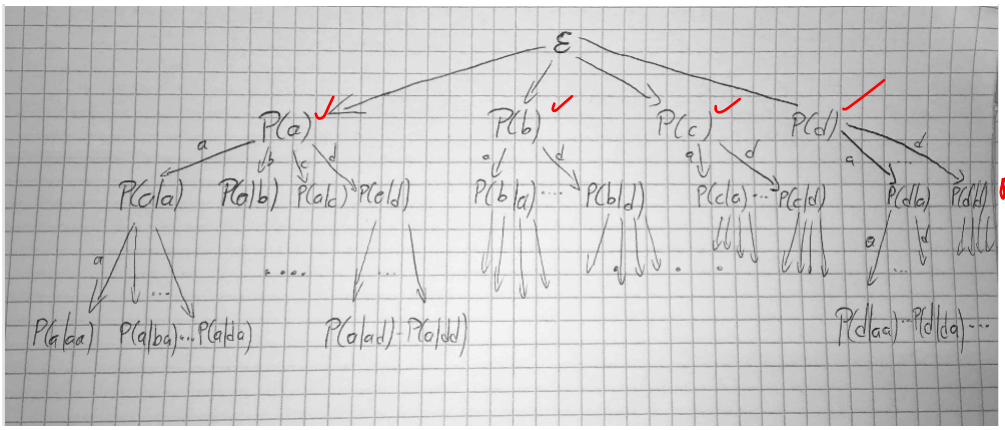
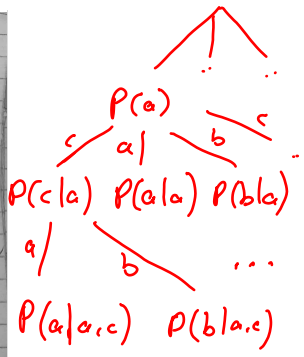
Statistical Natural Language Processing
Assignment 6

Sven Stauden 2549696
Janis Landwehr 2547715
Carsten Klaus 2554140

Exercise 1 1.5/2





The branching tree above represents a ngram model up to trigrams. To get the probability of a arbitrary n-gram, one has to traverse the tree to depth n. Let V be the size of a given vocabulary. As parameter we consider all the nodes in the branching tree excluding the root. For a n-gram model with vocabulary size V we need

$$\sum_{i=0}^{n} V^i - 1 \in O(V^n) \tag{1}$$

parameters to store the tree. This can be a major problem, because it takes an exponential amount of space. The larger V gets, the larger is the branching factor of the tree and the space consumption grows immensely.

Exercise 2 1/1

Kneser-Ney Smoothing is known as the state of the art technique for smoothing language model probabilities. The idea comes from absolute discounting,

Kneser-Ney =
obsolete discounting
+ observation of a surrounding scope

it basically adjusts the calculation low order ngrams, if higher order ngrams were never seen in the training data (or count is small). That means it considers backoffs. Higher Order n-grams work better but considering lower orders can assist the overall model. That means Kneser-Ney combines the approaches of absolute discounting and backoff language models.

Exercise 3 1/1

- The author's eloquence: Authors can be distinguished by their range of vocabulary, e.g. whether texts consist of rather easy / frequent words, or whether the author has a high level of learnedness.
- Texts can be classified by the author's syntax. Some authors prefer an easy to read style consisting of short sentences, while others try to connect logically dependent aspects, often resulting in long and nested texts with different long term dependencies.
- The morphological characteristics of an text can be linked to specific authors. This can be seen in texts that consist of many adjectives, where the authors focus on convey emotions and try to allow the readers to better immerse themselves deep into the story.
- Authors can be distinguished by the frequencies of specific words that are used. For example, some authors rather focus on specific topics or have a specific style on how they arouse specific emotions.
- In general, the language of the text is important for author recognition as usually authors do not publish texts in different languages (Translations can be seen as texts of a different author because the translator might uses own characteristics).

- tokens per line
- token length
- punctuation
- capital letters
- adjectives
- amount of specific words
- ...

as 0.5 only produce one plot per author

Exercise 4 & 5 2.5/2.5

See Python Notebook

Exercise 6 0.5/1

Unigram probabilities have a limited possibility to encode author characteristics which are useful for author classification. The frequency of author

▷ do not consider order of words
↳ negated sentences have equal probs:
P(This is not good but bad)
= P(This is bad but not good)

specific words, the language, etc. will be respected by the resulting model. A major problem is, that unigram probabilities are less dependent on the author but on the language itself. The high occurrence of the word "the" does (usually!) not depend on the author but on the English language. The probabilities of a subset of interesting words, text structural features or specific word combinations (n-grams) are less dependent on the language but on the way the author makes use of it, which is different from writer to writer and therefore more suitable for differentiation.

Exercise 7 1/1

A naive method to find out whether data is linear separable is to train a linear classifier on these data an check if a training accuracy of 100% can be achieved. If this is the case, the data is linear separable. Unregularized logistic regression but also the iterative Perceptron algorithm method introduced in the lecture would be sufficient.

Non linear separable data can be separated linearly by "bending" the vector space to a shape so that the points are separable after transformation. Neural networks achieve this by using non-linearity activation functions as ReLU. Support Vector machines solve this problem by using non-linear kernel functions like the Gaussian which maps the given data points to a more suitable vector space in which the data points are linear separable.

Exercise 8 0.5/0.5

While training language models we try to estimate the real probabilities of word frequencies (n-grams) in order to predict them for an unlabeled setting. The resulting parameters are probabilities that result from statistical consideration of training data.

Exercise 9 0.5/0.5

Minimizing the perplexity generally corresponds to increasing the prediction correctness of the language model / prediction model. A low perplexity results from high certainties of the language model which only appear if the model has really "learned" the correct estimation.

We are trying to maximize LL (log likelihood)! And afterwards we improve performance