Statistical Natural Language Processing
Assignment 4

Sven Stauden 2549696
Janis Landwehr 2547715
Carsten Klaus 2554140

Exercise 1

$$E_{new} = -0.5 \cdot log_2(0.5) - 0.5 \cdot log_2(0.5) = 1$$
$$E_{old} = -0.25 \cdot log_2(0.25) - 0.75 \cdot log_2(0.75) = 0.811$$
$$\text{Information Gain} = E_{new} - E_{old} = 1 - 0.811 = 0.189$$

Exercise 2

For non-equiprobable letters of an alphabet X some letters are likely to appear more often than others. A fix length encoding with a length l for each letter encoding would result to an average letter encoding length of $\sum_{x \in X} p(x) \cdot l$, where x is a letter from the alphabet.

Often occurring letters have a higher probability value than less often occurring ones. Therefore, reducing the length l for these often occurring letters also reduces the average encoding length.

In Morse encoding this ability is essential as shorter encodings especially mean shorter message transmission duration. One can observe, that often occurring letters like E (.), A (.-) or I (..) get encoded with short sequences while rare letters as X (-..-) or Q (–.-) have longer encodings.

Exercise 3

x/y	0	1	
0	1/4	1/4	P(x = 0) = 1/2
1	1/2	0	P(x = 1) = 1/2
	P(y = 0) = 3/4	P(y = 1) = 1/4	

Definition Conditional Entropy: $H(Y|X) = -\left(\sum_x P(x) \sum_y P(y|x) log(P(y|x)\right)$

$$\begin{aligned}
H(Y|X) = &-[P(X=0)(P(Y=0|X=0)log(P(Y=0|X=0))+ \\
&(P(Y=1|X=0)log(P(Y=1|X=0))))+ \\
&P(X=1)(P(Y=0|X=1)log(P(Y=0|X=1))+ \\
&(P(Y=1|X=1)log(P(Y=1|X=1))))] \\
= &-[\frac{1}{2}(\frac{1}{2} * log(\frac{1}{2}) + \frac{1}{2} * log(\frac{1}{2})) + \frac{1}{2}(1 * log(1))] \\
= &\frac{1}{2}
\end{aligned}$$

$$\begin{aligned}
H(X|Y) = &-[P(Y=0)(P(X=0|Y=0)log(P(X=0|Y=0))+ \\
&(P(X=1|Y=0)log(P(X=1|Y=0))))+ \\
&P(Y=1)(P(X=0|Y=1)log(P(X=0|Y=1))+ \\
&(P(X=1|Y=1)log(P(X=1|Y=1))))] \\
= &-[\frac{3}{4}(\frac{1}{3} * log(\frac{1}{3}) + \frac{2}{3} * log(\frac{2}{3})) + \frac{1}{4}(log(1))] \\
= &0.689
\end{aligned}$$

Definition Entropy: $-H(X) = \sum_x P(x) log P(x)$

$$\begin{aligned}
I(X|Y) &= H(Y) - H(Y|X) \quad \text{(see Venn diagram)} \\
H(Y) &= -[P(Y=0)logP(Y=0) + P(Y=1)logP(Y=1)] \\
&= -[\frac{3}{4}log(\frac{3}{4}) + \frac{1}{4}log(\frac{1}{4})] \\
&= 0.81 \\
I(X|Y) &= 0.81 - \frac{1}{2} \\
&= 0.31
\end{aligned}$$

log2 adds "bits"

better use $I(X;Y)$

shortcut: $I(X;Y) = H(X) + H(Y) - H(X,Y)$

Exercise 4

After one trial, the gambler wins $b_i \cdot T_i$ when he bet b_i of his current wealth and event i happened. We assume that this happens with a probability of p_i. Therefore the expectation of the rate of wealth increase is

$$E(RIW) = \sum_{i=1}^{n} p_i \cdot b_i \cdot T_i$$

. For the computation of the optimal distribution for the expected rate of increased wealth, we consider the logarithmic rate of wealth increase; the maximizers do not change thereby.

$$E(log_2(RIW)) = \sum_{i=1}^{n} p_i \cdot log_2(b_i \cdot T_i)$$

Further, the optimal distribution b does not change, when we discard T_i as in the following:

$$\begin{aligned} & argmax_b \sum_{i=1}^{n} p_i \cdot log_2(b_i \cdot T_i) \\ =& argmax_b \sum_{i=1}^{n} p_i \cdot log_2(b_i) + p_i \cdot log_2(T_i) \\ =& argmax_b \sum_{i=1}^{n} p_i \cdot log_2(b_i) \end{aligned}$$

Applying the KL-Divergence the following holds:

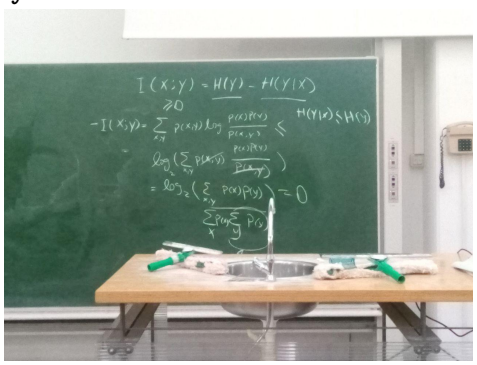
$$\begin{aligned} D(p||b) &= -\sum_{i=1}^{n} p_i \cdot log_2(b_i) - H(q) \\ \Leftrightarrow \sum_{i=1}^{n} p_i \cdot log_2(b_i) &= -H(q) - D(p||b) \end{aligned}$$

As $H(q)$ is fix, to maximize the left side of the last line w.r.t. to b, $D(p||b)$ has to be minimal. In slide 23 of chapter 4, we showed, that 0 is the minimal value of $D(p||b)$ and $D(p||b) = 0 \Leftrightarrow p = b$. Therefore the optimal choice of the wealth distribution for betting b is p. $\square$

Exercise 5 05/1

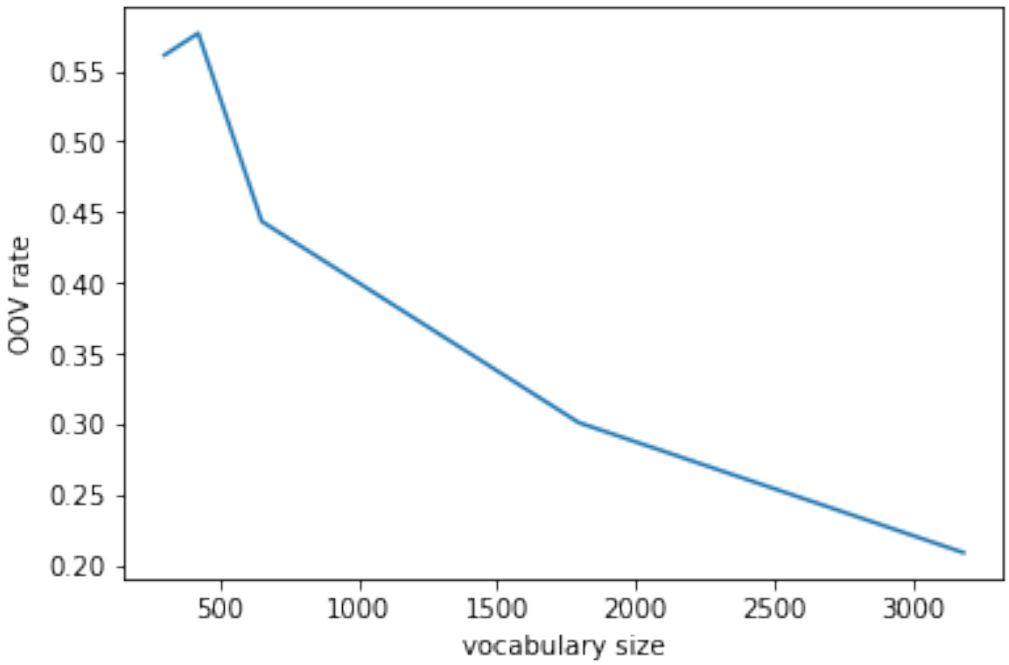
Proof

$$
\begin{aligned}
H(Y|X) &= \sum_x P(x) H(Y|X=x) \quad \texttt{Definition} \\
&= \sum_x P(x) \sum_y -P(y|x) log P(y|x) \quad \texttt{Definition} \\
&= \sum_y P(y) \sum_x -P(x|y) log P(y|x) \quad \texttt{since} P(y)P(x|y) = P(x)P(y|x) \\
&\leq \sum_y P(y) log \left(\sum_x \frac{P(x|y)}{P(y|x)} \right) \quad \texttt{Jensens inequality}^* \\
&= \sum_y P(y) log \left(\sum_x \frac{P(x)}{P(y)} \right) \quad \texttt{since} \\
&= \sum_y -P(y) log P(y) \quad \texttt{since} \sum_y P(y) = 1 \\
&= H(Y)
\end{aligned}
$$



* can be applied because for fixed y it holds that $\sum_x P(x|y) = 1$

Exercise 6 3/3



Explanation: When the vocabulary contains a small amount of words the probability that a word appears in a test text which is not in the vocabulary is higher then for vocabulary with a higher amount of words. Therefore, the larger the vocabulary size the lower will be the OOV value for a test set. ¡br¿¡br¿ OOV words might be probematic, as after learning a language model with training data, probability values are only computed for words in the (training) vocabulary. When computing a bigram probability in a certain test text it may happen that one of the two words is an OOV word and therefore does not have a trained probability i.e. its probability is 0. A 0-probability cannot be used for further computation. ¡br¿

A solution is to apply a smoothing method, which prevents 0-probabilities. Smoothing redistributes the probabilities of all words including OOV words and therefore manages that these words do not appear with probability 0.

Furthermore, applying back-offs could be applied i.e. one ignores the his-

tory sequence (in this case one word) and therefore transforms the bigram to a unigram and therefore prevents a zero-probability.

Implementation: See Python files

Exercise 7

Implementation: See Python files