EECS595: Natural Language Processing
Homework 1 Written Assignment

Zhihao Xu

September 13, 2021

1. Suppose the deletion and the insertion costs are "1" respectively and the substitution cost is "2". Figure out whether *drive* is closer to *brief* or to *divers*. What is the minimum edit distance in each case?

   Solution:

   MED(drive, brief) = 4

E	5	6	5	4	3	4
V	4	5	4	3	4	5
I	3	4	3	2	3	4
R	2	3	2	3	4	5
D	1	2	3	4	5	6
#	0	1	2	3	4	5
	#	B	R	I	E	F

Table 1: Minimum Edit Distance for drive and brief

   MED(drive, divers)=3

E	5	4	3	2	1	2	3
V	4	3	2	1	2	3	4
I	3	2	1	2	3	4	5
R	2	1	2	3	4	3	4
D	1	0	1	2	3	4	5
#	0	1	2	3	4	5	6
	#	D	I	V	E	R	S

Table 2: Minimum Edit Distance for drive and divers

   Hence *drive* is closer to *divers* than *brief*.

2. In the second lecture on N-grams, we introduced a very simple smoothing technique *Laplace Smoothing* (slide page 58). Please explain why once you add 1 to the count, the smoothed probabilities become

$$p_i^* = \frac{c_i + 1}{N + V}$$

(where V is the vocabulary size) and the discounted counts become

$$c_i^* = (c_i + 1) \times \frac{N}{N + V}$$

Solution:

For the smoothed probabilities, $p_i^* = \frac{c_i^{'}}{N'}$. After adding 1 to the count, $c_i^{'} = c_i + 1$ and $N' = \sum_{i=1}^{V} c_i^{'} = \sum_{i=1}^{V} c_i + 1 = N + V$. Hence $p_i^* = \frac{c_i^{'}}{N'} = \frac{c_i+1}{N+V}$.

In order to compute the smoothed probabilities, we need to change both the numerator and denominator, by defining an discounted count c^*, we can smooth the probability only change the numerator. In addition to adding 1, we also need to multiply c^* by a normalization factor $\frac{N}{N+V}$ to ensure $\sum_{i=1}^{V} c_i^* = N$. This can also be viewed as discounting non-zero counts in order to get the probability mass for the zero counts. For each non-zero count, the probability mass is lowered by $c_i - (c_i + 1)\frac{N}{N+V} = \frac{c_i V - N}{N+V}$, which will be assigned to the zero counts.

3. This problem is about Good-Turing smoothing which we did not cover in the class. Suppose we have a vocabulary V (i.e., a set of possible words). We'd like to estimate a unigram distribution $P(w)$ over $w \in V$. In the training set, we observe a total of N words. Note that this training set may not include all members in the vocabulary set particularly if N is small compared to $|V|$. For any word seen c times in the training sample, the Good-Turing estimate of its count is

$$c^* = \frac{(c + 1)N_{c+1}}{N_c}$$

where N_c is the number of members of V which are seen c times in the corpus. For any w which is observed in the training corpus, $P(w) = c^*(w)/N$, where $c^*(w)$ is the updated number of times w is seen in the training set based on GT-smoothing. Suppose $V^{'}$ represents the set of words that are not seen in the training set. Under this definition, prove that:

$$\sum_{w \in V'} P(w) = \frac{N_1}{N}$$

Hint: You can assume that c in the training data set ranges from 1 to m, and all $N_c, \forall c = 1, \cdots m$ is non-zero, and $N_{m+1} = 0$.

Solution:

By definition, $N = \sum_{c \geq 1} cN_c$

$$\begin{aligned}
\sum_{w \in V'} P(w) &= 1 - \sum_{w \in V} P(w) \\
&= 1 - \sum_{c \geq 1} \frac{c^*}{N} N_c \\
&= 1 - \sum_{c \geq 1} \frac{(c+1)N_{c+1}}{N_c N} N_c \\
&= \frac{N - \sum_{c \geq 1}(c+1)N_{c+1}}{N} \\
&= \frac{\sum_{c \geq 1} cN_c - \sum_{c \geq 1}(c+1)N_{c+1}}{N} \\
&= \frac{1 \times N_1}{N} = \frac{N_1}{N}
\end{aligned}$$