

Tutorial SNLP

February 1, 2020

1 Questions

Q1. Let the cost of insertion be 1, deletion be 2 and substitution be 3. Fill in the code snippet each of these cost in place of (A), (B), (C).

```
for i = 1,..., N do
  for j = 1,..., M do
    D(i, j) = min(D(i - 1, j) + (A) , D(i, j-1) + (B), D(i-1, j-1) + (C) )
  end for
end for
```

Find the edit distance between lead and deal.

Q2. While processing a corpus, you encounter a word “insention”. You recognize that this is a spelling error and also the possible candidate words are ‘insertion, inspection, invention, indention, intention’. Assuming that you are using a noisy channel model for spell correction, write down the expression for the correct word in terms of the probabilities to be estimated from a training corpus.

Q3. Corpus: b a b a a c b c a c a c

Vocabulary is $\{a, b, c\}$

We are considering a bi-gram language model, and a Good-Turing method for probability estimation. Use GT probabilities up to and including $r = 2$.

Q4.

We are given the following corpus, modified from the one in the chapter:

```
<s> I am Sam </s>
<s> Sam I am </s>
<s> I am Sam </s>
<s> I do not like green eggs and Sam </s>
```

Using a bigram language model with add-one smoothing, what is $P(\text{Sam} \mid \text{am})$? Include $\langle s \rangle$ and $\langle /s \rangle$ in your counts just like any other token.

Q5. From a restaurant corpus, it was found that the number of unique unigrams is 1500 and the total number of bigrams is 8000. Suppose out of 8000, 5000 bigrams occur once in the corpus, and rest of the bigrams are occurred twice. Use Good Turing smoothing to estimate the effective count for the bigrams not seen in the corpus. And also calculate the Probability mass that will be allotted to them.

Q6. Given a corpus C2, The Maximum Likelihood Estimation (MLE) for the bigram “ice cream” is 0.4 and the count of occurrence of the word “ice” is 310. The likelihood of “ice cream” after applying add-one smoothing is 0.025, for the same corpus C2. What is the vocabulary size of C and what is the likelihood after applying add-3 smoothing?

Q7. Consider the following corpus of two sentences.
One fish two fish red fish blue fish black fish blue fish
Once I caught a fish alive. The red fish turned blue
Assume you perform case-folding, punctuation removal and stop word removal on the corpus, before attempting any of the following questions.
(a) Write the count of types and tokens in the corpus.
(b) Show a table with the bigram counts (as in the text) for this corpus. Given this table, give the $P(\text{fish}|\text{two})$ and $P(\text{black}|\text{fish})$.
(c) Compute the probability mass that Good-Turing would assign to zero count bigrams, given this corpus.

Q8.

For the following corpus of two documents:

1. how much wood would a wood chuck chuck if a wood chuck would chuck wood
2. a wood chuck would chuck the wood he could chuck if a wood chuck would chuck wood

Which of the following sentences:

1. a wood could chuck;
2. wood would a chuck;

is more probable, according to:

1. An unsmoothed uni-gram language model?
2. A uni-gram language model, with Laplacian (“add-one”) smoothing?
3. An unsmoothed bigram language model?
4. A bi-gram language model, with Laplacian smoothing?
5. An unsmoothed tri-gram language model?
6. A tri-gram language model, with Laplacian smoothing?

Q9.

We are given the following corpus, modified from the one in the chapter:

```
<s> I am Sam </s>
<s> Sam I am </s>
<s> I am Sam </s>
<s> I do not like green eggs and Sam </s>
```

If we use linear interpolation smoothing between a maximum-likelihood bi-gram model and a maximum-likelihood unigram model with $\lambda_1 = \frac{1}{2}$ and $\lambda_2 = \frac{1}{2}$, what is $P(\text{Sam}|\text{am})$? Include `<s>` and `</s>` in your counts just like any other token.

Q10.

What does back-off mean, in the context of smoothing a language model? What does interpolation refer to? Why do we usually use log probabilities when finding the probability of a sentence according to an n-gram language model?

Q11. The backoff probability for a trigram LM is defined as:

$$\begin{aligned} P_{bo}(w_n|w_{n-1}w_{n-2}) &= P'(w_n|w_{n-1}w_{n-2}) \text{ if } c(w_{n-2}w_{n-1}w_n) > 0. \\ &= \lambda(w_{n-1}w_{n-2})P_{bo}(w_n|w_{n-1}), \text{ otherwise.} \end{aligned}$$

Similarly,


$$P_{bo}(w_n|w_{n-1}) = P'(w_n|w_{n-1}) \text{ if } c(w_{n-1}w_n) \text{ if } c(w_{n-1}w_n) > 0.$$

$$= \lambda(w_{n-1})P'(w_n), \text{ otherwise.}$$

$$P'(w_n|w_{n-1}) = P(w_n|w_{n-1}) - \frac{1}{8}$$

$$P'(w_n|w_{n-1}w_{n-2}) = P(w_n|w_{n-1}w_{n-2}) - \frac{1}{8}$$

Suppose your vocabulary contains a, b, c d and you are given the following counts:

b	a	5
b	b	3
b	c	0 
b	d	0

ab	a	4
ab	b	0
ab	c	0
ab	d	0

a	8
b	9
c	8
d	7

Construct the backoff LM $P_{bo}(w|w_{n-2}w_{n-1})$, where $w_{n-1} = b$ and $w_{n-2} = a$.

2 Solution

Q1. Insertion Cost:=1

Deletion Cost:=2

Substitution Cost:=3

So, the algorithm will be:

```

for j = 0,..., N do
    D(0, j) = j*1
end for
for i = 0,..., N do
    D(i, 0) = i*2
end for
for i = 1,..., N do
    for j = 1,..., M do
        if X[i] != Y[j] then
            D(i, j) = min(D(i - 1, j) + (2) , D(i, j-1) + (1), D(i-1, j-1) + (3) )
        else
            D(i, j) = min(D(i - 1, j) + (2) , D(i, j-1) + (1), D(i-1, j-1) + (0) )
        end if
    end for
end for

```

	#	D	E	A	L
#	0	1	2	3	4
L	2	3	4	5	3
E	4	5	3	4	5
A	6	7	5	3	4
D	8	6	7	5	6

Above is the table for computing edit distance between the words LEAD and DEAL.

Q2. Candidate words: insertion, inspection, invention, indention, intention.
Now, the noisy channel model says the correct word

$$\arg \max_{w \in V} P(w|x), \text{ where } x \text{ is the misspelled word}$$

$$\arg \max_{w \in V} P(x|w) \cdot p(w)$$

Remember that $P(x|w) = \frac{del[w_{i-1}, w_i]}{count[w_{i-1}, w_i]}$, if deletion

$$= \frac{ins[w_{i-1}, x_i]}{count[w_{i-1}]} , \text{ if insertion}$$

$$= \frac{subs[x_i, w_i]}{count[w_i]} , \text{ if substitution}$$

$$= \frac{trans[w_i, w_{i+1}]}{count[w_i, w_{i+1}]} , \text{ if transposition}$$

Let us write the expression of $P(x|w) \cdot P(w)$ for all w's.

$$P(\text{insertion} | \text{insertion}) \cdot P(\text{insertion}) = \frac{subs[n, r]}{count[r]} \cdot P(\text{insertion})$$

$$P(\text{insertion} | \text{inspection}) \cdot P(\text{inspection}) = \frac{subs[n, c]}{count[c]} \cdot \frac{del[s, p]}{count[s, p]} \cdot P(\text{inspection})$$

$$P(\text{insertion} | \text{invention}) \cdot P(\text{invention}) = \frac{subs[s, v]}{count[v]} \cdot P(\text{invention})$$

$$P(\text{insertion} | \text{indention}) \cdot P(\text{indention}) = \frac{subs[s, d]}{count[d]} \cdot P(\text{indention})$$

$$P(\text{insertion} | \text{intention}) \cdot P(\text{intention}) = \frac{subs[s, t]}{count[t]} \cdot P(\text{intention})$$

Q3.

Observed bigrams are $\{ba, ab, ba, aa, ac, cb, bc, ca, ac, ca, ac\}$

Unobserved bigrams: bb, cc

Observed bigram frequencies: ab: 1, aa: 1, cb: 1, bc: 1, ba: 2, ca: 2, ac: 3

$N_0 = 2, N_1 = 4, N_2 = 2, N_3 = 1, N = 11$

Will use GT probabilities up to and including $r = 2$

Probability estimations:

Use Good-Turing: $P(bb) = P(cc) = (0+1) \cdot (N_1 / (N \cdot N_0)) = 4 / (11 \cdot 2) = 2/11$

Use Good-Turing: $P(ab) = P(aa) = P(cb) = P(bc) = (1+1) \cdot (N_2 / (N \cdot N_1)) = 1/11$

Use Good-Turing: $P(ba) = P(ca) = (2+1) \cdot (N_3 / (N \cdot N_2)) = 3/22$

Q4.

A[i,j] = c(wi,wj)													
	I	am	Sam	and	do	not	like	green	eggs	<s>	</s>	Total	
I	0	3	0	0	1	0	0	0	0	0	0	4	
am	0	0	2	0	0	0	0	0	0	0	0	3	
Sam	1	0	0	0	0	0	0	0	0	0	0	3	4
and	0	0	1	0	0	0	0	0	0	0	0	0	1
do	0	0	0	0	0	1	0	0	0	0	0	0	1
not	0	0	0	0	0	0	1	0	0	0	0	0	1
like	0	0	0	0	0	0	0	1	0	0	0	0	1
green	0	0	0	0	0	0	0	0	1	0	0	0	1
eggs	0	0	0	1	0	0	0	0	0	0	0	0	1
<s>	3	0	1	0	0	0	0	0	0	0	0	0	4
</s>	0	1	3	0	0	0	0	0	0	0	0	0	4
													25

From the table above we can take the value of $c(\text{am}, \text{Sam}) = 2$ $c(\text{am}) = 3$.

$$P(\text{Sam}|\text{am}) = \frac{c(\text{am}, \text{Sam})+1}{c(\text{am})+|V| (=9)} = 0.25$$

Q5. Count of unique unigrams: 1500 = |V|

Total number of bigrams possible with these unigrams = $|V|^2$

Total number of unique bigrams observed = 8000. Now, we know what are the possible number of bigrams, we know how many bigrams are actually there. This means the difference between the possible number of bigrams and the number of unique bigrams gives the number of bigrams with count 0. It is already mentioned that $N_1=5000$.

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

put $c = 0$, $c^* = 0.0022$

$$P^*_{GT}(\text{Unobserved Bigrams}) = \frac{N_1}{N} = \frac{5000}{8000} = 0.625 \quad (2)$$

Q6. $\text{count}(\text{ice}) = 310$. $P(\text{cream}|\text{ice}) = 0.4$. So, from bigram probability, $\text{count}(\text{ice}, \text{cream}) = 310 * 0.4 = 124$

Now, after adding add-1 smoothing, the probability $\frac{\text{count}(\text{ice}, \text{cream})+1}{\text{count}(\text{ice})+|V|}$ which is given as 0.025.

From there, $|V| = 4690$. Now, after applying add-3 smoothing, the likelihood $\frac{\text{count}(\text{ice}, \text{cream})+3}{\text{count}(\text{ice})+3|V|} = 0.0088$

Q7. Stop words are i, a, the Type-11, if you consider one, two, once) Token-21

	<s>	alive	black	blue	caught	fish	once	one	red	turned	two
<s>	0	0	0	0	0	0	0	1	1	0	0
alive	0	0	0	0	0	0	0	0	0	1	0
black	0	0	0	0	0	0	1	0	0	0	0
blue	0	0	0	0	0	0	2	0	0	0	0
caught	0	0	0	0	0	0	1	0	0	0	0
fish	0	1	1	2	0	0	0	0	0	1	1
once	0	0	0	0	0	1	0	0	0	0	0
one	0	0	0	0	0	0	1	0	0	0	0
red	0	0	0	0	0	2	0	0	0	0	0
turned	0	0	0	1	0	0	0	0	0	0	0
two	0	0	0	0	0	1	0	0	0	0	0

$$c^* = \frac{(C+1)N_{c+1}}{N_c} \quad (3)$$

Here N_{c+1} is the count of bigrams that occur exactly one time, N_c is count of bigrams that occur exactly 0 times. Here c is zero (so ideally it should be occurring c times).

$$P_{GT}^*(things\ with\ frequency\ 0\ in\ training) = \frac{N_1}{N} \quad (3)$$

where

N_1 = count of things that were seen once in training, and

N = total number of things (bigrams) that actually occur in training

Q8.

So, the frequencies of the ten different word uni-grams: a (4), chuck (9), could (1),he (1),how (1), if (2),much (1), the (1), wood (8), would (4). Total: 32

$P(A) = P(a)P(wood)P(could)P(chuck)$, $\frac{4.8.1.9}{32.32.32.32} = 2.75 \cdot 10^{-4}$ $P(B) = P(wood)P(would)P(a)P(chuck)$, $\frac{8.4.4.9}{32.32.32.32} = 1.10 \cdot 10^{-3}$ So,probability of sentence 2 is more.

$$|V| = 10, p(a) = \frac{4+1}{32+10} c \quad P(A) = P(a)P(wood)P(could)P(chuck) = \frac{5.9.2.10}{42.42.42.42} = 2.89 \cdot 10^{-4} \quad P(B) = P(wood)P(would)P(a)P(chuck) = \frac{9.5.5.10}{42.42.32.32} = 7.23 \cdot 10^{-4}$$

Because we are considering bigrams here, we have to append the beginning of sentence and end of sentence marker. $P(A) = P(a|<s>)P(wood|a)P(could|wood)P(chuck|could)P(</s>|chuck)$ $\frac{1.4.0.1.0}{2.4.8.1.9} = 0$
 $P(B) = P(wood|<s>)P(would|wood)P(a|would)P(chuck|a)P(</s>|chuck) = \frac{0.1.1.0.0}{2.8.4.4.9} = 0$

$|V|=11$,as we are predicting $</s>$

$$P(A) = P(a|<s>)P(wood|a)P(could|wood)P(chuck|could)P(</s>|chuck) = \frac{2.5.1.2.1}{13.15.19.12.20} = 2.25 \cdot 10^{-5}$$

$$P(B) = P(wood|<s>)P(would|wood)P(a|would)P(chuck|a)P(</s>|chuck) = \frac{1.2.2.1.1}{13.19.15.15.20} = 3.60 \cdot 10^{-6}$$

Using two sentence terminals s_1 and s_2 here. $P(A) = P(a|<s_1><s_2>)P(wood|<s_2>a)P(<s_1>|chuck<s_2>)$ $\frac{2.1.0.0.0.0}{2.1.4.0.1.0}$ $P(B) = P(wood|<s_1><s_2>)P(would|<s_2>wood)P(<s_1>|chuck<s_2>)$ $\frac{0.0.1.0.0.0}{2.0.1.1.0.0}$

$$|V| = 12 \text{ because of } <s_1> \text{ and } <s_2> \quad P(A) = P(a|<s_1><s_2>)P(wood|<s_2>a)P(<s_1>|chuck<s_2>)$$

$$= \frac{2.2.1.1.1.1}{14.13.16.12.13.12} = 7.34 \cdot 10^{-7}$$

$$P(B) = P(wood|<s_1><s_2>)P(would|<s_2>wood)P(<s_1>|chuck<s_2>)$$

$$= \frac{1.1.2.1.1.1}{14.12.13.13.12.12} = 4.89 \cdot 10^{-7}$$

Q9.

A[i,j] = c(wi,wj)													
	I	am	Sam	and	do	not	like	green	eggs	<s>	</s>	Total	
I		0	3	0	0	1	0	0	0	0	0	0	4
am		0	0	2	0	0	0	0	0	0	0	1	3
Sam		1	0	0	0	0	0	0	0	0	0	3	4
and		0	0	1	0	0	0	0	0	0	0	0	1
do		0	0	0	0	0	1	0	0	0	0	0	1
not		0	0	0	0	0	0	1	0	0	0	0	1
like		0	0	0	0	0	0	0	1	0	0	0	1
green		0	0	0	0	0	0	0	0	1	0	0	1
eggs		0	0	0	1	0	0	0	0	0	0	0	1
<s>		3	0	1	0	0	0	0	0	0	0	0	4
</s>		0	1	3	0	0	0	0	0	0	0	0	4
													25

From the table above we can take the value of $c(\text{am}, \text{Sam}) = 2$.

$$\text{Bigram Probability: } P(\text{Sam}|\text{am}) = \frac{c(\text{am}, \text{Sam})}{c(\text{am})} = 0.66$$

$$\text{Unigram Probability: } P(\text{Sam}) = \frac{c(\text{Sam})}{\text{No_of_tokens}} = 0.12$$

$$\text{Interpolated Probability} = 0.5 * 0.66 + 0.5 * 0.12 = 0.39$$

Q10. a) Back-off is a different smoothing strategy, where we incorporate lower-order n-gram models (in particular, for unseen contexts). For example, if we have never seen some tri-gram from our sentence, we can instead consider the bigram probability (at some penalty, to maintain the probability of all of the events, given some context, summing to 1). If we haven't seen the bi-gram, we consider the uni-gram probability. If we've never seen the uni-gram (this token doesn't appear in the corpus at all), then we need a so-called "0-gram" probability, which is a default for unseen tokens.

Using Log avoids underflow, and adding is faster than multiplying.

Q11.

In this question, a few counts are given. Eg. ab a 4, b a 5, etc. Read these counts as number of times aba has occurred is 4, number of times ba has occurred is 5, etc.

$$P'(w_n | w_{n-1}) = P(w_n | w_{n-1}) - \left(\frac{1}{8}\right)$$

$$P'(w_n | w_{n-1} w_{n-2}) = P(w_n | w_{n-1} w_{n-2}) - \left(\frac{1}{8}\right)$$

$$\begin{aligned} \text{Therefore, } P'(b|b) &= P(b|b) - \frac{1}{8} \\ &= \frac{3}{3+5} - \frac{1}{8} = \frac{2}{8} \quad (\text{Refer to the 1st table for the counts}) \end{aligned}$$

$$P'(a|b) = \frac{5}{3+5} - \frac{1}{8} = \frac{4}{8}$$

$$P'(c|b) = P'(d|b) = 0$$

$$\begin{aligned} \text{Now, } P_{bo}(w_n | w_{n-1}) &= P'(w_n | w_{n-1}) \text{ if } c(w_{n-1} w_n) \text{ if } c(w_{n-1} w_n) > 0. \\ &= \lambda(w_{n-1}) P'(w_n), \text{ otherwise.} \end{aligned}$$

$$\text{So, } P_{bo}(a|b) = P'(a|b)$$

$$P_{bo}(b|b) = P'(b|b)$$

$$P_{bo}(c|b) = \lambda(b).P(c), \quad P_{bo}(d|b) = \lambda(b)P(d)$$

$$\text{Now, } \sum P_{bo}(w_n|b) = 1$$

$$\text{So, } P_{bo}(a|b) + P_{bo}(b|b) + P_{bo}(c|b) + P_{bo}(d|b) = 1 \dots (i)$$

$$\text{Now, } \sum P_{bo}(w_n|b) = 1$$

$$\text{So, } P_{bo}(a|b) + P_{bo}(b|b) + P_{bo}(c|b) + P_{bo}(d|b) = 1 \dots (i)$$

$$\text{Again, } P(c) \text{ (look at the 3rd table given in the question)} = 8 / 32$$

$$P(d) = 7 / 32$$

$$\text{So, from (i), } \frac{4}{8} + \frac{2}{8} + \lambda(b) \cdot \frac{8}{32} + \lambda(b) \cdot \frac{7}{32} = 1$$

$$\text{From the above equation we have } \lambda(b) = \frac{8}{15}$$

$$\text{That makes } P_{bo}(c|b) = \frac{2}{15}, \quad P_{bo}(d|b) = \frac{7}{60}$$

$$\begin{aligned} \text{Next, given : } P_{bo}(w_n | w_{n-1} w_{n-2}) &= P'(w_n | w_{n-1} w_{n-2}), \text{ if } c(w_{n-2} w_{n-1} w_n) > 0. \\ &= \lambda(w_{n-1} w_{n-2}) P_{bo}(w_n | w_{n-1}), \text{ otherwise.} \end{aligned}$$

$$\text{Therefore, } P_{bo}(a | ba) = P'(a | ba) = P(a | ba) - \frac{1}{8} = \frac{4}{4} - \frac{1}{8} = \frac{7}{8}$$

$$P'(b | ba) = P'(c | ba) = P'(d | ba) = 0$$

$$\text{Again, } \sum P_{bo}(w_n|ba) = 1$$

$$\text{So, } P_{bo}(b | ba) + P_{bo}(c | ba) + P_{bo}(d | ba) = 1 - P_{bo}(a | ba)$$

$$\begin{aligned} \lambda(ba) [P_{bo}(b | b) + P_{bo}(c | b) + P_{bo}(d | b)] &= 1 - \frac{1}{8} \\ &= \frac{7}{8} \end{aligned}$$

$$\text{From the above equation, we can easily calculate } \lambda(ba) = \frac{1}{4}$$

$$P_{bo}(b | ba) = P_{bo}(b | b) \cdot \lambda(ba) = \frac{1}{16}$$

$$P_{bo}(c | ba) = P_{bo}(c | b) \cdot \lambda(ba) = \frac{1}{30}$$

$$P_{bo}(d | ba) = P_{bo}(d | b) \cdot \lambda(ba) = \frac{7}{240}$$