Department of Computing

Imperial College of Science, Technology and Medicine

example

example description

Author: Anton Zhitomirsky

Contents

1 Sparsity

1.1 Problem

If certain words are absent then the proability in the n-gram is zero. We can use smoothing to solve this or give it unknown word tokens.

1.2 Add-one smoothing

	i	want	to	eat	chinese	food	lunch	spend
i	6	828	1	10	1	1	1	3
want	3	1	609	2	7	7	6	2
to	3	1	5	687	3	1	7	212
eat	1	1	3	1	17	3	43	1
chinese	2	1	1	1	1	83	2	1
food	16	1	16	1	2	5	1	1
lunch	3	1	1	1	1	2	1	1
spend	2	1	2	1	1	1	1	1

(a) Counts after one-smoothing (we've added +1 to EVERYTHING)

	i	want	to	eat	chinese	food	lunch	spend
i	0.002	0.33	0	0.0036	0	0	0	0.00079
want	0.0022	0	0.66	0.0011	0.0065	0.0065	0.0054	0.0011
to	0.00083	0	0.0017	0.28	0.00083	0	0.0025	0.087
eat	0	0	0.0027	0	0.021	0.0027	0.056	0
chinese	0.0063	0	0	0	0	0.52	0.0063	0
food	0.014	0	0.014	0	0.00092	0.0037	0	0
lunch	0.0059	0	0	0	0	0.0029	0	0
spend	0.0036	0	0.0036	0	0	0	0	0

(b) probabilities before one-smoothing

	i	want	to	eat	chinese	food	lunch	spend
i	0.0015	0.21	0.00025	0.0025	0.00025	0.00025	0.00025	0.00075
want	0.0013	0.00042	0.26	0.00084	0.0029	0.0029	0.0025	0.00084
to	0.00078	0.00026	0.0013	0.18	0.00078	0.00026	0.0018	0.055
eat	0.00046	0.00046	0.0014	0.00046	0.0078	0.0014	0.02	0.00046
chinese	0.0012	0.00062	0.00062	0.00062	0.00062	0.052	0.0012	0.00062
food	0.0063	0.00039	0.0063	0.00039	0.00079	0.002	0.00039	0.00039
lunch	0.0017	0.00056	0.00056	0.00056	0.00056	0.0011	0.00056	0.00056
spend	0.0012	0.00058	0.0012	0.00058	0.00058	0.00058	0.00058	0.00058

(c) probabilities after one-smoothing

i	want	to	eat	chinese	food	lunch	spend
2533	927	2417	746	158	1093	341	278

(d) total instances of words

Figure 1: difference between smoothing and not

$$P_{add-1}(w_n|w_{n-1}) = \frac{C(w_{n-1}, w_n) + 1}{C(w_{n-1}) + V}$$

Given words with sparse statistics, they steal probability mass from more frequent words. This is because smaller instances of vocabulary are more influenced by the V and +1 term appearing in the new fraction. However, for larger occurrences they shrink because the nominator is large (so +1) won't change it by much, but the denominator grows larger, making the overall probability less.

The change here is great between the hilighted terms in Figure 1. "If we have very few instances of the word 'want' the smoothing will impact this a lot and this number will therefore change a lot. Wheras, if we have a lot of counts within the word 'i' then it wont change the probability a lot." Indeed, looking at the total instances of word i, the denominator is so large already that the number wasn't impacted, and 'want' was impacted.

1.2.1 Problems

Although easy to implement, it takes too much probability mass from more likely occurrences (see Figure 1) and assigns too much probability to unseen events. Therefore, we could try +k smoothing witha smaller value of k.

1.3 Back-off smoothing

If we don't have any occurrences in our current (trigram) model, we can back-off and see how many occurrences there are in the smaller (bigram) model.

Definition 1.1 (Back off smoothing "sutpid back-off")**.**

$$S(w_i|w_{i-2}w_{i-1}) = \begin{cases} \frac{C(w_{i-2}w_{i-1}w_i)}{C(w_{i-2}w_{i-1})}, & if\ C(w_{i-2}w_{i-1}w_i) > 0 \\ 0.4 \cdot S(w_i|w_{i-1}), & otherwise \end{cases}$$

$$S(w_i|w_{i-1}) = \begin{cases} \frac{C(w_{i-1}w_i)}{C(w_{i-1})}, & if\ C(w_{i-1}w_i) > 0 \\ 0.4 \cdot S(w_i), & otherwise \end{cases}$$

$$s(w_i) = \frac{C(w_i)}{N}$$

Where N is the number of words in the text

1.3.1 Problems

suppose that the bigram "a b" and the unigram "c" are very common, but the trigram "a b c" is never seen. Since "a b" and "c" are very common, it may be significant (that is, not due to chance) that "a b c" is never seen.

Perhaps it's not allowed by the rules of the grammar. Instead of assigning a more appropriate value of 0, the method will back off to the bigram and estimate $P(c|b)$, which may be too high

1.4 Interpolation

Combine evidence from different n-grams:

Definition 1.2 (Interpolation)**.**

$$P_{interp}(w_i|w_{i-2}w_{i-1}) = \lambda_1 P(w_i|w_{i-2}w_{i-1}) + \lambda_2 P(w_i|w_{i-1}) + \lambda_3 P(w_i), \quad where\ \lambda_1 + \lambda_2 + \lambda_3 = 1$$