



N-gram smoothing

Linguistics 684.02



Smoothing

- Handle **the sparse data problem**
- Called smoothing, because they make probabilities more equal
- Smoothing is most necessary for probabilities that are estimated from very few samples, least necessary for those that are based on lots of data



Lidstone or Laplace

- Idea, seen last time, is to add in some measure of uniform distribution

$$P(w[i] | w[i-1]) = \frac{\lambda + |w[i-1], w[i]|}{\sum_j \lambda + |w[i-1]w[j]|}$$



Backoff and interpolation

- An n-gram model is a conditional distribution of $P(w[i] | w[i-1], w[i-2], \dots)$
- Problem is that the condition on the distribution makes it rare or even absent. For n-grams, the condition is often called **the history**
- Shared idea, use lower-order conditional distributions
 - In **backoff**, we use the higher-order distributions when they exist, otherwise we use the lower-order ones
 - In **interpolation**, both the higher and the lower order distributions are used, even when higher-order distributions exist. Mixing coefficients ensure that the conditional distributions sum to 1.



Deleted interpolation

$$P_I(w[i] | w[i-1]) = \lambda P(w[i] | w[i-1]) + (1 - \lambda) P(w[i])$$



Deleted interpolation

- The power of this approach is that each history can have a different λ
 - Focus on the higher-order distribution when history is common
 - Focus on the lower-order distribution when history is rare
- The challenge of this approach is estimating the λ s
 - This is a tough problem if each λ is estimated separately
 - A sensible approach is to estimate λ s by grouping histories together in some way.
 - This is called **parameter-tying**, and is a generally useful device.



Good Turing smoothing

- Goal: adjust the estimates so as to allocate some probability to items that have zero count in the observed corpus
- Idea: group the items that were observed by frequency



Frequency	Count	Total
1	120	120
2	54	108
3	37	111
4	24	96
5	16	80
...		



r	Nr	rNr	r^*
0	?		
1	120	120	
2	54	108	
3	37	111	
4	24	96	
5	16	80	
...			



Backoff

$$P_{Backoff}(event) = \begin{cases} test_1(event) \rightarrow P_1(event) \\ test_2(event) \rightarrow P_2(event) \\ test_3(event) \rightarrow P_3(event) \\ \dots \end{cases}$$



Backoff

- Here, I'm assuming that the tests are applied in order and are mutually exclusive



Backoff

$$P_{Backoff}(w[n] | w[n-1]) = \begin{cases} C(w[n], w[n-1]) > k \rightarrow \dots \\ 1 \leq C(w[n], w[n-1]) \leq k \rightarrow \dots \\ 0 \leq C(w[n], w[n-1]) \rightarrow \dots \\ \dots \end{cases}$$