

050/520/600.666 Information Extraction from Speech and Text

Homework # 9

Due May 8, 2006.

Comparison of Held-Out and Good-Turing Estimates

We will compare the held-out estimate of Section 15.2 and the Good-Turing estimate of Section 15.4 by using *text A* and *text B* form our projects. In particular, we will use text A to develop our probability estimates, and text B to check their empirical performance in terms of the *average log-likelihood* or perplexity. For this task, the alphabet \mathcal{X} will be *non-overlapping pairs of consecutive letters*, so that $|\mathcal{X}| = 729$, $|A| = 15,000$ and $|B| = 2,500$. We will assume that consecutive symbols are independent, and estimate *unigram* probability distributions on \mathcal{X} .

1. Let the first 12,000 symbols in text A be the development set \mathcal{D} , and the remaining 3,000 the held-out set \mathcal{H} . Use the procedure described in Section 15.2 to
 - (a) begin with a provisional value of $M = 20$;
 - (b) use the counts $c_d(x)$ in \mathcal{D} to create the equivalence classes $\Phi : \mathcal{X} \rightarrow \{0, 1, \dots, M+1\}$, i.e $\Phi(x) = i$ if and only if $c_d(x) = i$;
 - (c) use the counts r_i in \mathcal{H} to estimate the class-probabilities λ_i , $i = 0, 1, \dots, M+1$;
 - (d) use the counts $c_d(x)$ in \mathcal{D} to get the class membership counts n_i , $i = 0, 1, \dots, M$, the relative frequency estimates $f_d(x)$ for x with $\Phi(x) = M+1$, and the probability P_M ;
 - (e) use the considerations in Section 15.2.3 to choose an appropriate value of M in Step 1a above, then repeat Steps 1b through 1d;
 - (f) compute the probability estimate

$$\tilde{P}(x) = \begin{cases} \lambda_i \times \frac{1}{n_i} & \text{if } \Phi(x) = i \in \{0, 1, \dots, M\}, \\ \lambda_{M+1} \times \frac{f_d(x)}{P_M} & \text{if } \Phi(x) = M+1, \end{cases} \quad x \in \mathcal{X}, \quad (1)$$

and verify that it sums to unity;

- (g) compute the perplexity of text B using the held-out estimate of equation (1);
- (h) (*alternative*) just before Step 1f, pool the data-sets \mathcal{D} and \mathcal{H} back together and use the counts $c_d(x)$ in $\mathcal{D} \cup \mathcal{H}$ to recalculate the equivalence classes $\Phi : \mathcal{X} \rightarrow \{0, 1, \dots, M+1\}$, the class membership counts n_i , the estimates $f_d(x)$ and P_M in (1).

Compare the performance of the estimate with and without the optional Step 1h of merging together \mathcal{D} and \mathcal{H} for reestimating the class membership and relative frequencies.

2. Good-Turing estimation does not require dividing the text A into \mathcal{D} and \mathcal{H} . Therefore,

- (a) begin with a provisional value of M as determined in Step 1e above;
- (b) compute the counts $c_d(x)$, equivalence classes $\Phi : \mathcal{X} \rightarrow \{0, 1, \dots, M + 1\}$, the class membership counts n_i , the estimates $f_d(x)$ and P_M from the entire text A;
- (c) using N to denote $|A|$, compute the probability estimate

$$\hat{P}(x) = \begin{cases} \frac{(i+1)n_{i+1}}{n_i N} & \text{if } \Phi(x) = i \in \{0, 1, \dots, M\}, \\ \alpha f_d(x) & \text{if } \Phi(x) = M + 1, \end{cases} \quad x \in \mathcal{X}, \quad (2)$$

where α is computed, as described in Section 15.4, to ensure that $\hat{P}(\cdot)$ sums to unity.

- (d) compute the perplexity of text B using the Good-Turing estimate of equation (2).

How should one choose M in Step 2a above? Experiment with a few different values.

Compare the perplexity of the Good-Turing estimate with that of the held-out estimates of Part 1.

Investigation of Katz' Back-off Formula

We will implement the Katz back-off formula of Section 15.7, again using text A and text B from our projects. Our alphabet will once again be *non-overlapping pairs of consecutive letters*, so that $|\mathcal{X}| = 729$, $|A| = 15,000$ and $|B| = 2,500$. However, we will build a trigram model for this alphabet, so that we will have to deal, at least conceptually, with an alphabet $\mathcal{X} \times \mathcal{X} \times \mathcal{X}$ of size 387,420,489 for the Good-Turing estimate $P_T(w_1, w_2, w_3)$ in equation (24) on page 271, and an alphabet $\mathcal{X} \times \mathcal{X}$ of size 531,441 for the corresponding bigram estimate required for computing $Q_T(w_3|w_2)$ in equation (23). The estimate (2) above will play the role of $f(w_3)$ in equation (23).

1. Begin by creating bigram counts $c_d(\langle w_2, w_3 \rangle)$ for all seen bigrams $\langle w_2, w_3 \rangle \in \mathcal{X} \times \mathcal{X}$. *Unlike* the way you converted text A & B from lower-case letters to symbols of \mathcal{X} by chunking them into nonoverlapping pairs-of-letters, here you should extract overlapping bigrams $\langle w_2, w_3 \rangle$. i.e., you should extract $N - 1$ bigram tokens, where $N = 15,000$ is the length of text A.

- (a) Provisionally choose $K = 7$ for bigrams;
- (b) Using the counts $c_d(\cdot)$ obtained above, compute the Good-Turing estimate $P_T(w_2, w_3)$ as a function of the bigram count $c_d(\langle w_2, w_3 \rangle)$, for $0 \leq c_d(\langle w_2, w_3 \rangle) < 7$;
- (c) Check that for all $i \in \{0, \dots, K - 1\}$, and any two bigrams $\langle w_2, w_3 \rangle$ and $\langle w'_2, w'_3 \rangle$,

$$c_d(\langle w_2, w_3 \rangle) = i \quad \text{and} \quad c_d(\langle w'_2, w'_3 \rangle) = i + 1 \quad \Rightarrow \quad P_T(w_2, w_3) \leq P_T(w'_2, w'_3),$$

and, if necessary, revise your choice of K in Step 1a to achieve this;

- (d) Using the counts-of-counts, $n_i =$ the number of bigrams with count $c_d(\langle w_2, w_3 \rangle) = i$, compute the coefficient

$$\alpha = \frac{\sum_{i=2}^{K-1} i n_i}{\sum_{i=2}^K i n_i};$$

- (e) For each seen “history” w_2 , use the probability estimates of *infrequent but seen* bigrams from Step 1b, the α from Step 1d, and the relative frequency estimates of *frequent* bigrams to compute

$$\beta(w_2) = \frac{1}{\sum_{\langle w_2, w_3 \rangle \in \mathcal{S}_0} \hat{P}(w_3)} \left[1 - \alpha \sum_{\langle w_2, w_3 \rangle \in \mathcal{S}^*} Q_T(w_3|w_2) - \sum_{\langle w_2, w_3 \rangle \in \mathcal{S}_K} f_d(w_3|w_2) \right],$$

where the sums range over $w_3 \in \mathcal{X}$ for a fixed w_2 , $Q_T(w_3|w_2) = \frac{P_T(\langle w_2, w_3 \rangle)}{f_d(w_2)}$,

$$\begin{aligned} \mathcal{S}_0 &= \{\langle w_2, w_3 \rangle : c_d(\langle w_2, w_3 \rangle) = 0\} && \text{are the unseen bigrams,} \\ \mathcal{S}^* &= \{\langle w_2, w_3 \rangle : 1 \leq c_d(\langle w_2, w_3 \rangle) < K\} && \text{are the infrequent seen bigrams,} \\ \mathcal{S}_K &= \{\langle w_2, w_3 \rangle : c_d(\langle w_2, w_3 \rangle) \geq K\} && \text{are the frequent bigrams,} \end{aligned}$$

and $\hat{P}(w_3)$ in the denominator is the Good-Turing estimate (2) we developed earlier.

- (f) Compute the back-off bigram estimate

$$\hat{P}(w_3|w_2) = \begin{cases} f_d(w_3|w_2) & \text{if } c_d(\langle w_2, w_3 \rangle) \geq K, \\ \alpha Q_T(w_3|w_2) & \text{if } 1 \leq c_d(\langle w_2, w_3 \rangle) < K, \\ \beta(w_2) \hat{P}(w_3) & \text{if } c_d(\langle w_2, w_3 \rangle) = 0, \end{cases} \quad (3)$$

and check that it sums to unity for each *seen* history w_2 .

Explain how the formula in equation (3) generalizes for *unseen* histories w_2 .

2. Repeat the exercise above with trigram counts $c_d(\langle w_1, w_2, w_3 \rangle)$, and compute the trigram estimate $\hat{P}(w_3|w_1, w_2)$ of equation (22) in Section 15.7 on page 271.
3. Compute the perplexity of text B for the bigram and trigram models of Parts 1 and 2.

Survey of Smoothing Techniques for Language Modeling

Read the following report and summarize it in your own words in no more than 4 pages.

S. Chen and J. Goodman, “An Empirical Study of Smoothing Techniques for Language Modeling,” *Harvard University Technical Report* TR-10-98, August, 1998.

In particular, comment on the formulae for various smoothing techniques studied by the authors.