Homework #3: CMPT-413

Due in class on Feb 13, 2003

Anoop Sarkar – anoop@cs.sfu.ca

(1) (150pts) This homework will use two data files: `atis3.pos` (the ATIS part of speech tagged corpus) and `wsj3_00.pos` (Section `0` from the Penn Treebank Wall Street Journal part of speech tagged corpus). The file `wsj3_00.pos` will be our source of *training data* and the file `atis3.pos` will be our source of *test data*.

a. (20pts) Use the supplied Perl program `clean-tagged.pl` to create the files `wsj3_00.pos.tags` and `wsj3_00.pos.emit` from `wsj3_00.pos`. Also, create the files `atis3.pos.tags` and `atis3.pos.emit` from `atis3.pos`.

Then use the supplied Perl programs `skip.pl`, `paste.pl`, `clean-ngram.pl`, `ngramCounts.pl` and `ngramLogProb.pl` to produce a bigram model of the part of speech tag sequences from the file `wsj3_00.pos.tags` (this is our training data). Before computing the bigram counts using `ngramCounts.pl`, remove unwanted bigrams using `clean-ngram.pl`.

The output of `ngramLogProb.pl` will be a bigram model of tag sequences stored as log probabilities: $\log_2 P(t_i \mid t_{i-1})$.

b. (80pts) Let $T = s_0, \ldots, s_m$ represent the test data with sentences s_0 through s_m.

$$P(T) = \prod_{i=0}^{m} P(s_i) = 2^{\sum_{i=0}^{m} \log_2 P(s_i)}$$

$$\log_2 P(T) = \sum_{i=0}^{m} \log_2 P(s_i)$$

where $\log_2 P(s_i)$ is the log probability assigned by the bigram model to the sentence s_i (note that in this homework, each s_i is a sequence of part of speech tags). These log probabilities are provided by `ngramLogProb.pl` (see Question 1a). Let W_T be the length of the text T measured in part of speech tags. The *cross entropy* for T is:

$$H(T) = -\frac{1}{W_T} \log_2 P(T)$$

The cross entropy corresponds to the average number of bits needed to encode each of the W_T words in the *test data*. The *perplexity* of the test data T is defined as:

$$PP(T) = 2^{H(T)}$$

Write a Perl program to compute the cross entropy and perplexity for a given input file. The input to the program is a bigram model over part of speech tag sequences and an input file with sentences that are part of speech tag sequences.

Using this program, print out the cross entropy and perplexity for the training data `wsj3_00.pos.tags` and the test data `atis3.pos.tags`.

On the test data, when a bigram in unseen, the probability for that bigram is zero. However, since we are using log probabilities, we cannot use a probability of zero (as $\log_2(0)$ is not defined). Instead, use the value $\log_2 P(t_i \mid t_{i-1}) = -99999$, when a bigram (t_{i-1}, t_i) is unseen.

Remember that cross entropy and perplexity are both positive real numbers, and the lower the values, the better the model over the test data.

c. (50pts) Implement the following Jelinek-Mercer style *interpolation* smoothing model:

$$P_{interp}(t_i \mid t_{i-1}) = \lambda P(t_i \mid t_{i-1}) + (1 - \lambda)P(t_i)$$

Set $\lambda = 0.8$ and using P_{interp} recompute the cross-entropy and perplexity for the training data `wsj3_00.pos.tags` and the test data `atis3.pos.tags`. Provide the output of the program. Can you find a value for λ that results in a better model of the test data?

Use the simplifying assumption that if the unigram t_i is unseen then $\log_2 P(t_i) = -99999$.

— OR —

d. (50pts) Implement add-one smoothing to provide counts for every possible bigram (t_{i-1}, t_i). Using `ngramLogProb.pl` and your Perl program from Question 1b, recompute the cross-entropy and perplexity for the training data `wsj3_00.pos.tags` and the test data `atis3.pos.tags`. Provide the output of the program. Do not smooth the unigram model $P(t_i)$.

For extra credit, you can do *both* Question 1c and 1d. Once both methods are implemented, the bigram probability $P(t_i \mid t_{i-1})$ in P_{interp} can be replaced by the add-one smoothing model to reduce test set perplexity even further.