CSEP 517, Fall 2015: Assignment 1

Due: Monday, October 19th at 5pm

This assignment contains a writing and a programming question, which will also requires a written report of experimental results. Your final, combined writeup should be no more than four pages long. Good luck!

1 Problem 1 (5pt)

In class, we discussed how back off can improve the performance of language models. Consider the following back-off scheme. First, we define the sets

$$
\begin{aligned}
\mathcal{A}(w_{i-1}) &= \{w : c(w_{i-1}, w) > 0\} \\
\mathcal{B}(w_{i-1}) &= \{w : c(w_{i-1}, w) = 0\} \\
\mathcal{A}(w_{i-2}, w_{i-1}) &= \{w : c(w_{i-2}, w_{i-1}, w) > 0\} \\
\mathcal{B}(w_{i-2}, w_{i-1}) &= \{w : c(w_{i-2}, w_{i-1}, w) = 0\}
\end{aligned}
$$

where c is a function that counts n-grams in the training set. For example, if the bigram "Honey Bunny" appears 22 times in the corpus, we will have $c(\mathit{Honey}, \mathit{Bunny}) = 22$.

Now, we can define a back-off trigram model:

$$
p(w_i|w_{i-2}, w_{i-1}) = \begin{cases} p_1(w_i|w_{i-2}, w_{i-1}) & \text{If } w_i \in \mathcal{A}(w_{i-2}, w_{i-1}) \\ p_2(w_i|w_{i-2}, w_{i-1}) & \text{If } w_i \in \mathcal{A}(w_{i-1}) \text{ and } w_i \in \mathcal{B}(w_{i-2}, w_{i-1}) \\ p_3(w_i|w_{i-2}, w_{i-1}) & \text{If } w_i \in \mathcal{B}(w_{i-1}) \end{cases}
$$

Where:

$$
\begin{aligned}
p_1(w_i|w_{i-2}, w_{i-1}) &= p_{ML}(w_i|w_{i-2}, w_{i-1}) \\
p_2(w_i|w_{i-2}, w_{i-1}) &= \frac{p_{ML}(w_i|w_{i-1})}{\sum_{w \in \mathcal{B}(w_{i-2}, w_{i-1})} p_{ML}(w|w_{i-1})} \\
p_3(w_i|w_{i-2}, w_{i-1}) &= \frac{p_{ML}(w_i)}{\sum_{w \in \mathcal{B}(w_{i-1})} p_{ML}(w)}
\end{aligned}
$$

Questions

1. *(2 points)* Does the above model form a valid probability distribution? Prove your answer.
2. *(3 points)* If it does not form a valid probability distribution, suggest how to make it one by modifying p_1, p_2 and p_3, using p_{ML} (the maximum likelihood estimate) and/or the count function c, and briefly explain why your modification works.

2 Problem 2 (10pt)

In this programming problem, you will build and evaluate two language models. We provide three corpora to conduct the evaluation, each from a different domain. The data files are formatted with each line containing a tokenized sentence (white spaces mark token boundaries). In addition, each corpus is accompanied by a README file describing its origin. Please, split each dataset into training (80%), validation (development) (10%) and test (10%).

NOTE: You can use the programming language of your choice. However, please provide well commented code if you want any partial credit. Additionally, if you have multiple files, please provide a short description in the preamble of each file. Your submission will not be evaluated for efficiency, but we recommend keeping such issues in mind to better streamline the experiments.

Questions

1. *(3 points)* Implement the language model you discussed in Problem 1. If you found the given language model to be an invalid probability distribution, please use the correct one you suggested.
2. *(3 points)* Implement another language model that uses linear interpolation between unigram, bi-gram, and tri-grams models.
3. *(1 point)* Report the perplexities of the two language models using the different corpora.
4. *(1 point)* What hyper-parameters (i.e. λ_1, λ_2, λ_3) did you use for each dataset in the linear interpolation model? Briefly describe your approach for choosing them.
5. Now we will explore issues of transferring your model between domains. Please, use the linear interpolation model for this question.
   a. *(1 point)* Report the perplexities for the following combinations:
      Train on Reuters and test on Brown.
      Train on Brown and test on Gutenberg.
      Train on Gutenberg and test on Reuters.
   b. *(1 point)* Based on the previous results, how does transferring a model from a source domain to a different target domain influence performance? What does it tell you about the language used in the domains and their similarity? Provide graphs, tables, charts or other summary evidence to support any claims you make.

Bonus (2pt) Design an approach for using a small fraction of the target domain data to adapt the model trained on the source domain. How does it influence performance? How close can you get to within-domain performance?