NLP Homework 1

Question 1:

The following feature sets could help solve the OOV problem:

- Suffix features, e.g.,
$$f_{101}(x,y) = \begin{cases} 1 & x\ ends\ with\ \text{ing}\ and\ y = VBG \\ 0 & o.w. \end{cases}$$
- Prefix features, e.g.,
$$f_{102}(x,y) = \begin{cases} 1 & x\ starts\ with\ \text{pre}\ and\ y = NN \\ 0 & o.w. \end{cases}$$
- Contextual features, e.g.,
$$f_{106}(x_i,y_i) = \begin{cases} 1 & previous\ word\ w_{i-1} = \text{the}\ and\ y = Vt \\ 0 & o.w. \end{cases}$$

Question 2:

a. There are 10^5 words and 25 possible tags, hence there are $(10^5)^2$ possible combinations for the two previous words and $25 \cdot 10^5$ possibilities for the current word and it's label.
Therefore, the total number of possible feature features is $25 \cdot (10^5)^3$.
b. As seen above, the number of possible feature combinations is extremely large, therefore training on huge corpora would allow us to encounter most of the relevant ones and effectively estimate the distribution parameters.
However, a reasonably sized corpus would probably incur a significant bias on the MLE parameters.

Question 3:

Exponential function is used in this formulation for the following reasons:

- Smooths the probability distribution such that no (x,y) pair receives a null probability.
- Differentiable, which is useful for optimization.
- More sensitive to differences in the distribution i.e., more significant differences between high and low probabilities.

Question 4:

We would use 100,000 binary features since we would like all features to receive the same initial weight since the vocabulary is ordered in an arbitrary way. However, assigning integer values to each feature could create scaling problems since the first word receives the value 1 and the last word 100,000.

In addition, we would like each bit in the representation to correspond to a different feature in order to weight the features individually. Using a single integer feature makes this

impossible under the given method since different integers use the same bits, therefore the weight assigned to one directly affect the others.

Question 5:

The problem with this formulation is that $|\tilde{X}|$ (the number of possible feature vectors over input examples) is very large, meaning that we need to perform an unreasonable number of computations in order to calculate the probabilities of this model.

Question 6:

We avoid the problem encountered in question 5 by summing over the label set instead of the feature vector set, very significantly reducing the number of calculations needed to find each probability.

Question 7:

While HMM models are better at generating (word, tag) sequences, MEMM models are much better at classification (POS tagging) tasks.