Question 1 (20 points)

Clarissa Linguistica decides to build a log-linear model for language modeling. She has a training sample \((x_i, y_i)\) for \(i = 1 \ldots n\), where each \(x_i\) is a prefix of a document (e.g., \(x_i = \text{“Yesterday, George Bush said”}\)) and \(y_i\) is the next word seen after this prefix (e.g., \(y_i = \text{“that”}\)). As usual in log-linear models, she defines a function \(f(x, y)\) that maps any \(x, y\) pair to a vector in \(\mathbb{R}^d\). Given parameter values \(v \in \mathbb{R}^d\), the model defines

\[
P(y|x, v) = \frac{e^{v \cdot f(x, y)}}{\sum_{y' \in \mathcal{V}} e^{v \cdot f(x, y')}}
\]

where \(\mathcal{V}\) is the vocabulary, i.e., the set of possible words; and \(v \cdot f(x, y)\) is the inner product between the vectors \(v\) and \(f(x, y)\).

Given the training set, the training procedure returns parameters \(v^* = \arg \max_v L(v)\), where

\[
L(v) = \sum_i \log P(y_i|x_i, v) - C \sum_k |v_k|^2
\]

and \(C > 0\) is some constant.

Clarissa makes the following choice of her first two features in the model:

\[
f_1(x, y) = \begin{cases} 1 & \text{if } y = \text{model and previous word in } x \text{ is the} \\ 0 & \text{otherwise} \end{cases}
\]

\[
f_2(x, y) = \begin{cases} 1 & \text{if } y = \text{model and previous word in } x \text{ is the} \\ 0 & \text{otherwise} \end{cases}
\]

So \(f_1(x, y)\) and \(f_2(x, y)\) are identical features.

Question (10 points): Show that for any training set, with \(f_1\) and \(f_2\) defined as above, the optimal parameters \(v^*\) satisfy the property that \(v_1^* = v_2^*\).

Question (10 points): Now say we define the optimal parameters to be \(v^* = \arg \max_v L(v)\), where

\[
L(v) = \sum_i \log P(y_i|x_i, v) - C \sum_k |v_k|
\]

and \(C > 0\) is some constant. (Here \(|v_k|\) is the absolute value of the \(k\)’th feature.) In this case, does the property \(v_1^* = v_2^*\) necessarily hold? If not, what constraints do hold for the values \(v_1^*\) and \(v_2^*\)?

Question 2 (15 points)

Nathan L. Pedant now decides to build a bigram language model using log-linear models. He gathers a training sample \((x_i, y_i)\) for \(i = 1 \ldots n\). Given a vocabulary of words \(\mathcal{V}\), each \(x_i\) and each \(y_i\) is a member of
Each \((x_i, y_i)\) pair is a bigram extracted from the corpus, where the word \(y_i\) is seen following \(x_i\) in the corpus.

Nathan’s model is similar to Clarissa’s, except he chooses the optimal parameters \(v^*\) to be \(\arg \max L(v)\) where

\[
L(v) = \sum_i \log P(y_i|x_i, v)
\]

The features in his model are of the following form:

\[
f_i(x, y) = \begin{cases} 
1 & \text{if } y = \text{model and } x = \text{the} \\
0 & \text{otherwise}
\end{cases}
\]

i.e., the features track pairs of words. To be more specific, he creates one feature of the form

\[
f_i(x, y) = \begin{cases} 
1 & \text{if } y = w_2 \text{ and } x = w_1 \\
0 & \text{otherwise}
\end{cases}
\]

for every \((w_1, w_2)\) in \(\mathcal{V} \times \mathcal{V}\).

**Question (15 points):** Assume that the training corpus contains all possible bigrams: i.e., for all \(w_1, w_2 \in \mathcal{V}\) there is some \(i\) such that \(x_i = w_1\) and \(y_i = w_2\). The optimal parameter estimates \(v^*\) define a probability \(P(y = w_2|x = w_1, v^*)\) for any bigram \(w_1, w_2\). Show that for any \(w_1, w_2\) pair, we have

\[
P(y = w_2|x = w_1, v^*) = \frac{\text{Count}(w_1, w_2)}{\text{Count}(w_1)}
\]

where \(\text{Count}(w_1, w_2) = \text{number of times } (x_i, y_i) = (w_1, w_2)\), and \(\text{Count}(w_1) = \text{number of times } x_i = w_1\).

**Question 3**

Nathan L. Pedant generates \((x, y)\) pairs as follows. Take \(\mathcal{V}\) to be set of possible words (vocabulary), e.g., \(\mathcal{V} = \{\text{the, cat, dog, happy, \ldots}\}\). Take \(\mathcal{V}'\) to be the set of all words in \(\mathcal{V}\), plus the reversed string of each word, e.g., \(\mathcal{V}' = \{\text{the, eht, cat, tac, dog, god, happy, yppah, \ldots}\}\).

For each \(x\), Nathan chooses a word from some vocabulary \(\mathcal{V}\). He then does the following:

- With 0.4 probability, he chooses \(y\) to be identical to \(x\).
- With 0.3 probability, he chooses \(y\) to be the reversed string of \(x\).
- With 0.3 probability, he chooses \(y\) to be some string that is neither \(x\) nor the reverse of \(x\). In this case he chooses \(y\) from the uniform distribution over words in \(\mathcal{V}'\) that are neither \(x\) nor the reverse of \(x\).

**Question (10 points)**

Define a log-linear model that can model this distribution \(P(y|x)\) perfectly (Note: you may assume that there are no palindromes in the vocabulary, i.e., no words like \(\text{eye}\) which stay the same when reversed.) Your model should make use of as few parameters as possible (we will give you 10 points for a correct model with 2 parameters, 8 points for a correct model with 3 parameters, 5 points for a correct model with more than 3 parameters.)
Question (10 points)

Write an expression for each of the probabilities

\[ P(\text{the}|\text{the}) \]
\[ P(\text{eh}|\text{the}) \]
\[ P(\text{dog}|\text{the}) \]

as a function of the parameters in your model.

Question (10 points)

What value do the parameters in your model take to give the distribution described above?