Questions

"Training"

1. How are word vectors trained?
2. Describe what it means to use a one-hot embedding representation of a categorical feature.
3. Consider the word "cat". Let v be the 50-dimensional GloVe embedding of the word "cat", and let w be the 100-dimensional GloVe embedding of the same word. Do the first 50 entries of w corresponds to the entries of v? Why or why not?
4. When should you use `nn.LSTM` rather than `nn.RNN`?
5. Explain how to pad sequences to form a mini-batch to train an RNN.

"Generalization"

1. Consider embedding a tweet by summing the GloVe embeddings of the words that appear in the tweet. Suppose that you would like to measure "how similar" two tweets are, using this tweet embedding. Would you use the cosine similarity or the Euclidean distance?
2. Character-level RNNs are RNNs that takes in a text input one character (one letter) at a time. What are advantages of a character-level RNN? What are disadvantages of a character-level RNN?
3. Can you use word vectors alongside Character-level RNNs?
4. There are several issues with our implementation of `TweetBatcher`. For one, it is problematic to choose a random minibatch by first randomly choosing a tweet length and then choosing a batch with that sequence length. Why?