ECE467 – Pset 3

Jonathan Lam

May 10, 2021

1. Consider a neural language model (NLM) that uses a softmax layer as its output layer. How many nodes where there be in the output layer? What is the typical interpretation of the value of each output node?

   NLM is somewhat of a categorization problem to determine the most probable next word. Thus, there would be as many nodes as the size of the vocabulary, with the interpretation of the output being a probability distribution over how likely it is that each word comes next.

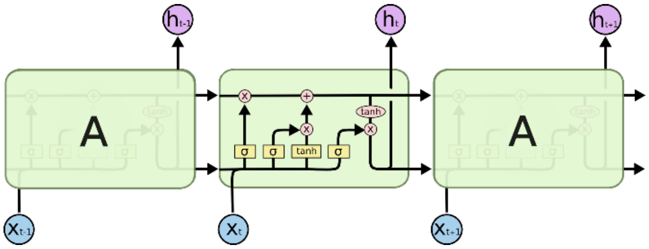
2. Assume that EMB(w) is the word embedding for a word, w, computed by some common technique for computing word embeddings (e.g., word2vec). Consider the vector obtained by the following vector arithmetic:

   EMB("Japan") + EMB("Paris") - EMB("Tokyo")

   Other than the words that already related to this equation, what word's embedding would likely be close to the result? Briefly explain (in one sentence) your answer.

   Hopefully, "France" should be close: the intuition is that vector embeddings hold semantic information in their numeric components, so we can apply "vector math" to analogies. (We can think of this, very loosely, as starting with the country Japan, adding the country France and the idea of a capital, and then subtracting out the idea of Japan and its capital; this leaves us with the France component.)

3. Consider the enrolled depiction of a few LSTM cells below, which was discussed in class and was taken from https://colah.github.io/posts/2015-08-Understanding-LSTMs. What is represented by the output of each of the three sigmoid functions in the cell (please limit your responses to one sentence each).



   The sigmoid functions each act as a gate to decide what information to let through. From left to right, the first is the "forget gate," which decides which parts of the context to throw away; the second is the "input gate," which decides which parts of the context to update; and the third is the "output gate," which decides what to emit to the cell's hidden state.

4. We have learned that encoder-decoder networks (a.k.a. sequence-to-sequence models) can be useful for certain NLP tasks, such as machine translation (MT). For MT, the encoder produces a context vector, which can be thought of as representing the input sentence from a source language, while the decoder generates a predicted translation in the target language. Briefly explain (at a high level, using one or two sentences) why adding attention to such a model can be useful.

   In seq2seq models where there isn't a one-to-one mapping from input to output (nor is the mapping usually in order, as different languages tend to have very different sentence structure), we need a way to choose a (fixed-length) context vector that is representative of the meaning of the sentence. Training the decoder with attention allows it to use a more complex function of the hidden states that may be able to produce more fluent translations (in the case of MT) or other decodings.

5. Briefly explain (in one sentence) one major reason why producing embeddings for characters or subwords instead of, or in addition to, full words can be useful.

   It may often be unrealistic to learn embeddings for each word in a language, because there are simply too many (and some words may be extremely rare) but morphological parts of a word (subwords) tend to be much more common; this does well on out-of-vocabulary prediction.