HW 11

March 28, 2023

1 Info

Please answer the following questions. They are based on Lecture 8. They are all written, no code.

2 Problems

Problem 1 In the BiDAF model, the authors discuss p^{start} and p^{end} the start and end token probabilities (in the paper, these are p^1 and p^2). From the setup, these are each dimension T, the length of the input sentence. A good model would put a the highest probability on $p^{start}_{y_{start}} p^{end}_{y_{end}}$, the probability of the question spanning $[start, end]$ indices in the passage (see Problem 4). Assume these are optimized for and you want to find $k < l$ such that $p^1_k p^2_l$ is maximized; i.e. you want to find the highest probability span which would be the answer to the question you posed. Describe a $O(T^2)$ algorithm to find the optimal (k, l) pair. Describe a $O(T)$ algorithm.

Problem 2 Some people might argue that there is some sort of attention in ELMo. What weights might they be referring to? Why?

Problem 3 What does COVE's text classification methodology (see lecture) do when there is only one sentence? What is an example of an NLP task that has 2 sentences and asks if they logically follow? What is one popular dataset for such a task?

Problem 4 What is special about the SQUAD data set in terms of the questions and the passages?

Problem 5 Here are some questions on ULM-Fit.

- Describe the 3 steps of ULM-Fit at a high level.
- What do the authors argue should be the representation fed to each classifier? I.e. What is the input to the new classifier layer added in Step 3?
- What is catastrophic forgetting? What is discriminative fine tuning in ULM-Fit?
- What is gradual unfreezing in ULM-Fit?

Problem 6 Suppose we use Hierarchical softmax as in Lecture 8: split the token vocabulary V into c clusters $\{V_1, \ldots, V_c\}$ of roughly equal size K and randomly assign words to 1 cluster each. Suppose that word j (j is the integer mapping of some string) is in cluster r and we are interested in computing $P(w_{t+1} = j | w_t, \ldots, w_1)$.

1 What is the complexity to compute softmax for a vocabulary of size $|V|$? I.e. If we just used softmax, what is the complexity of $P(w_{t+1} = j | w_t, \ldots, w_1)$?

2 Argue why $P(w_{t+1} = j | w_t, \ldots, w_1) = P(w_{t+1} = j, j \in V_r | w_t, \ldots, w_1)$. The "event" $j \in V_r$ is the event that we are considering cluster V_r. Remember the assumption of the location of j above.

3 Argue why

$$P(w_{t+1} = j | w_t, \ldots, w_1, j \in V_r) = P(w_{t+1} = j, | w_t, \ldots, w_1, j \in V_r) P(j \in V_r | w_t, \ldots, w_1)$$

4 We have $c * K = |V|$ by assumption. Given this, what should be the choice of c and K so that we compute Hierarchical softmax as fast as possible? Prove this.