Natural Language Processing (CS22N)

Student Name:

Babak Behkamkia

Instructor Name:

Dr. Sauleh Eetemadi

Spring 2023

1. **Attention exploration**

a.

i. We use softmax to calculate the alpha values. Thus, these values will be a number between 0 and 1 and their sum would be 1. This shows these values are a probability distribution. Moreover, each alpha value assigned to a word in the sequence and shows the importance of that word in that sequence of words. Therefore, alpha scores are a categorical probability distribution.

ii. $k_i q >> k_j q$ where $i, j \in \{1, 2, ..., n\}$ and $i != j$

iii. In this case, the value of c would be very close to v_i because $k_i q$ has the largest value and after applying the softmax function, the corresponding value would be 1 for it. Moreover, from the equation (1) we can understand that $c = v_i$.

iv. If the key value of a word is similar to q value of the chosen word, the product of those matrices will be large and after softmax, it will put almost all the weight on the corresponding alpha.

b.

i)

① * All basis vectors have norm 1 and are orthogonal to each other.

② * The two sub spaces are orthogonal.

③ $\begin{cases} v_a = c_1 a_1 + c_2 a_2 + \cdots + c_m a_m \\ v_b = c_1 b_1 + c_2 b_2 + \cdots + c_p b_p \end{cases}$

①, ③ $\Rightarrow a_i^T v_a = a_i^T c_1 a_1 + \cdots + a_i^T c_i a_i + \cdots + a_i^T c_m a_m$

$\Rightarrow a_i^T v_a = c_i \quad , \text{for } i \in \{1, 2, \ldots, m\}$

②, ③ $\Rightarrow a_i^T v_b = 0 \quad , \text{for } i \in \{1, 2, \ldots, m\}$

* same for b_j , for $j \in \{1, 2, \ldots, p\}$

$M = [a_1 a_1^T, a_2 a_2^T, \ldots, a_m a_m^T]$

$\rightarrow M s = M(v_a + v_b) = c_1 a_1 + c_2 a_2 + \cdots + c_m a_m$

ii)

$$c \approx \frac{1}{2}(v_a + v_b) \Rightarrow \begin{cases} \alpha_a = \frac{1}{2} \\ \alpha_b = \frac{1}{2} \end{cases}$$

$$\Rightarrow k_a^T q \approx k_b^T q \gg k_i$$

$$q = B(k_a + k_b), \quad B \rightarrow \text{large number}$$

$$\Rightarrow \begin{cases} k_a^T q = B \\ k_b^T q = B \\ k_i^T q = 0 \end{cases} \rightarrow \text{they are orthogonal}$$

$$\rightarrow \alpha_a = \alpha_b = \frac{\exp(B)}{n-2+2\exp(B)} \approx \frac{\exp(B)}{2\exp(B)} = \frac{1}{2}$$

c.

c)

i)

$$\left.\begin{array}{l} k_i \sim N(\mu_i, \Sigma_i) \\ \Sigma_i = \alpha I \end{array}\right\} k_i \sim N(\mu_i, \alpha I)$$

$$\Rightarrow k_i \approx u_i \longrightarrow q = B(\mu_a + \mu_b)$$

ii)

$$\Sigma_i = \alpha I + \frac{1}{2}(\mu_a \mu_a^T)$$

$$k_a \sim N\left(\mu_a, \alpha I + \frac{1}{2}(\mu_a \mu_a^T)\right)$$

$$\Rightarrow k_a \approx \beta \mu_a \quad \text{and} \quad \beta \sim N\left(1, \frac{1}{2}\right).$$

$$q = B(\mu_a + \mu_b) \Rightarrow \begin{cases} q^T \cdot k_a = \beta B \\ q^T \cdot k_b = B \\ \text{o.w.} \quad 0 \end{cases}$$

$$c = \sum_{i=1}^{n} v_i \alpha_i \approx \frac{\exp(\beta B)}{\exp(\beta B) + \exp(B)} v_a + \frac{\exp(B)}{\exp(\beta B) + \exp(B)} \cdot v_b$$

$\rightarrow$ if β grows $\rightarrow$ c is closer to v_a

d.

d)

i)

$$\begin{cases} q_1 = \beta_1 \mu_a \\ q_2 = \beta_2 \mu_b \end{cases} \leftarrow k_a \sim \mu_a$$

ii)

① $\begin{cases} c_1 \sim v_a \\ c_2 \sim v_b \end{cases}$

② $c = \frac{1}{2}(c_1 + c_2)$

①, ② $\rightarrow c = \frac{1}{2}(v_1 + v_2)$

2. Pretrained Transformer models and knowledge access

d.
   Accuracy: 2%
   London accuracy: 5%

f. Accuracy: 26.4%

g.
   i. Accuracy: 13%

3. Considerations in pretrained knowledge

a. Overall, pretrained models are better than non-pretrained models because they have trained on a large dataset previously. In other words, their weights are not random and they have learned some knowledge from the previous task.

b.
   1. Trustworthiness: If an application generates unreal information like made up birthplaces, research papers, or websites, it will reduce the users trust on the application.
   2. Using wrong information: People may not notice the made up answer and use it in critical situations, which leads to a big problem.

c. Obviously, the model cannot determine the birthplace of a person that it never seen it before but with providing more information about that person for the model can help it in order to find similar individuals and make a prediction based on this information. For example, being angry most of the time may be exclusive to the people of a specific country.