Homework 1 Language Models and HMM

Xinyi Zhao

October 12, 2023

1 Q 2

1. $\theta(v|u) = \frac{count(u,v)+1}{count(u)+|V|}$, and because $count(u,v)+1 > 0$ and $count(u)+|V|$¿0, so $\theta(v|u) > 0$.

   And because $count(u,v) \leq count(u), 1 \leq |V|$, so $\theta(v|u) \leq 1$

   $$\begin{aligned}\sum_v \theta(v|u) &= \frac{\sum_v count(u,v) + |V|}{count(u) + |V|} \\ &= \frac{count(u) + |V|}{count(u) + |V|} \\ &= 1\end{aligned}$$

   So it's a probability measure for all u.

2. $\theta(v|u) = \frac{count(u,v)+k}{count(u)+k|V|}$, and because $count(u,v)+k > 0$ and $count(u)+k|V|$¿0, so $\theta(v|u) > 0$.

   And because $count(u,v) \leq count(u), k \leq k|V|$, so $\theta(v|u) \leq 1$

   $$\begin{aligned}\sum_v \theta(v|u) &= \frac{\sum_v count(u,v) + k|V|}{count(u) + k|V|} \\ &= \frac{count(u) + k|V|}{count(u) + k|V|} \\ &= 1\end{aligned}$$

   So it is a probability measure for all u.

3. $\theta(v|u) = \frac{count(u,v)+k\theta(v)}{count(u)+k}, k > 0$. Because $count(u,v) + k\theta(v) > 0$ and $count(u) + k > 0$, so $\theta(v|u) > 0$.

Because $count(u,v) \leq count(u), k\theta(v) \leq k$, so $\theta(v|u) \leq 1$

$$\begin{aligned}\sum_v \theta(v|u) &= \frac{\sum_v count(u,v) + k\sum_v \theta(v)}{count(u)+k} \\ &= \frac{count(u)+k}{count(u)+k} \\ &= 1\end{aligned}$$

So it's a probability measure for all u.

4. since $\lambda_i \geq 0$, and each θ^* is a probability distribution (non-negative), so the linear combination is also non-negative. $\theta(w|v,u) = \lambda_1\theta^*(w|v,u) + \lambda_2\theta^*(v|u) + \lambda_3\theta^*(v)$

$$\begin{aligned}&\sum_w \theta(w|v,u) = \lambda_1 \sum_w \theta^*(w|v,u) + \lambda_2 \sum_w \theta^*(w|u) + \lambda_3 \sum_w \theta^*(w) \\ &= \lambda_1 \sum_w \frac{count(u,v,w)}{count(u,v)} + \lambda_2 \sum_w \frac{count(u,w)}{count(u)} + \lambda_3 \sum_w \frac{count(w)}{count(*)} \\ &= \lambda_1 + \lambda_2 + \lambda_3 \\ &= 1\end{aligned}$$

and the other terms are probability distributions (sum to 1), this implies that the sum of $\theta(w|v,u)$ is indeed 1. So it is a probability distribution for any (u, v),

$perplexity = 2^{NL/T}$. it is a linear interpolation of probability distributions. $\theta^*(v)$ will not be zero, so the overall model does not assign zero probability to any sequence, thus avoiding infinite perplexity.

5. Since each$\lambda_i^b > 0$ and each θ^* is a probability distribution (hence non-negative), the linear combination is also non-negative.

$$\begin{aligned}&\sum_{w\in V} \theta(w|v,u) = \sum_{w\in V} \left(\lambda_1^{\Pi(u,v)}\theta^*(w|v,u) + \lambda_2^{\Pi(u,v)}\theta^*(w|u) + \lambda_3^{\Pi(u,v)}\theta^*(w)\right) \\ &= \lambda_1^{\Pi(u,v)} + \lambda_2^{\Pi(u,v)} + \lambda_3^{\Pi(u,v)} \\ &= 1\end{aligned}$$

So it's a probability measure for all (u,v).

6. Since a(u), θ^* are non-negative, $\theta^\beta(v|u)$ is also non-negative.

$$\sum_v \theta(v|u) = \sum_{v \in A(u)} \frac{count^\beta(u,v)}{count(u)} + \sum_{v \in B(u)} a(u) \frac{\theta^\star(v)}{\sum_{w \in B(u)} \theta^\star(w)}$$
$$= \sum_{v \in A(u)} \frac{count^\beta(u,v)}{count(u)} + a(u)$$
$$= \sum_{v \in A(u)} \frac{count^\beta(u,v)}{count(u)} + 1 - \sum_{v \in A(u)} \frac{count^\beta(u,v)}{count(u)}$$
$$= 1$$

So the sum of $\theta(v|u)$ for any u is 1. This is a probability measure for each u.

2 Q 3

The perplexity of the Language Model on the Development set is given by: $2^{-\frac{N}{T}\frac{1}{N}\sum_i^N log(p_{xi})}$.

To minimize the perplexity, we need to minimize: $-\sum_i^N log(p_{xi})/N$.

Notice that this expression is the negative of the average log likelihood. The perplexity function is a monotonically decreasing function of the likelihood. As the likelihood increases, the perplexity decreases, and vice versa. Therefore, minimizing this expression is equivalent to maximizing the average log likelihood, which is our original likelihood function $L(\lambda_1, \lambda_2, \lambda_3)$.

3 Q 4

since $\lambda_i \geq 0$, $\sum \lambda_i^b = 1$. and each θ^* is a probability distribution (non-negative). What is wrong if we do this? Is theta(w—v,u) a probability measure for all (v,u)?

$$\sum_{w \in V} \theta(w|v,u) = \sum_{w \in V} \left(\lambda_1^{\Pi(u,v,w)} \theta^*(w|v,u) + \lambda_2^{\Pi(u,v,w)} \theta^*(w|u) + \lambda_3^{\Pi(u,v,w)} \theta^*(w) \right)$$

The issue arises because each term in the summation is a weighted sum of different conditional probabilities, and these weights $\lambda_i^\pi(w,v,u)$ are dependent on the specific (u,v,w) and vary each term in the summation, preventing the sum from equating to 1.

To fix this, he weights λ_i are chosen such that they are constant for a given (v,u) pair when summing over w. That is $\pi(v,u)$ instead of $\pi(w,v,u)$.

4 Q 5

Initialize a table D, where D[t][x] stores the maximum possibility on ending in x and the corresponding sequence

Algorithm 1 Maximum-probability-sequence(θ^*,T,V)

initialize table D of size $T \times |V|$
D[0][START]=1, and D[0][x]=0 for all other x in V.
while $t < T$ **do**
  for $x \in V$ **do**
    $D[t][x] = max_{v \in V} D[t-1][v] \times \theta^*(x|v)$
  end for
  Store u that maximize $D[t][u]$ with the probability.
  t+=1
end while
$D[T][STOP] = max_{v \in V} D[T-1][v] \times \theta^*(STOP|v)$
return the reconstructed sequences and their probability.

For all the time set, calcuate the probability for all the (x,v) pair. So the running time of the recursion step is $O(T|V^2|)$

5 Q 6

Suppose we have training data where we have 100 examples with x1 = a, x2 =b and y1 = A, y2 = B, 100 examples with x1 = a, x2 = c and y1 = A, y2 = C and 800 examples with x1 = c, x2 = d and y1 = B, y2 = D. Suppose we use a Bigram HMM Tagger on this data. What are the estimated parameters? Also, what are the highest probability sequences y we get when we tag (a, c) and (c, d).

We have $V = \{a, b, c, d\}$ and $K = \{A, B, C, D\}$
We observe 100 times x = (a, b) generating y = (*,A,B,STOP)
We observe 100 times x = (a,c) generating y = (*,A,C,STOP)
We observe 800 times x = (c,d) generating y = (*,B,D,STOP)
We have: count(START) = 1000, count(START, A) = 200,
count(START,B) = 800, count(A) = 200, count(A,B) = 100, count(A,C) = 100,
count(B) = 900, count(B, STOP) = 100, count(B, D) = 800,
count(C) = 800, count(C,STOP) = 800, count(D) = 800, count(D, STOP) = 800

We need to consider:
Transition Probabilities:$\theta(y_i|y_{i-1})$ and emission Probabilities: $\vartheta(x_i|y_i)$

$$\theta(A|\text{START}) = \frac{count(\text{START}, A)}{count(\text{START})} = 1/5, \theta(B|\text{START}) = 4/5, \theta(B|A) = 1/2,$$
$$\theta(C|A) = 1/2, \theta(D|B) = 8/9, \theta(STOP|B) = 1/9,$$
$$\theta(STOP|C) = 1, \theta(STOP|D) = 1, \vartheta(a|A) = 1,$$
$$\vartheta(b|B) = 1/9, \vartheta(c|B) = 8/9, \vartheta(c|C) = 1, \vartheta(d|D) = 1$$

$$\begin{aligned}
P(a, c, A, C) &= \theta(A|START)\theta(C|A)\theta(STOP|C)\vartheta(a|A)\vartheta(c|C) \\
&= 1/5 \cdot 1/2 \cdot 1 \cdot 1 \cdot 1 = 1/10 \\
p(a, c, A, B) &= \theta(A|START)\theta(B|A)\theta(STOP|B)\vartheta(a|A)\vartheta(c|B) \\
&= 1/5 \cdot 1/2 \cdot 1/9 \cdot 1 \cdot 8/9 = 4/405 \\
p(c, d, B, D) &= \theta(B|START)\theta(D|B)\theta(STOP|D)\vartheta(c|B)\vartheta(d|D) \\
&= 4/5 \cdot 8/9 \cdot 1 \cdot 8/9 \cdot 1 = 256/405 \\
p(c, d, C, D) &= \theta(C|START)\theta(D|C)\theta(STOP|D)\vartheta(c|C)\vartheta(d|D) = 0
\end{aligned}$$

So the highest probability sequences when we tag (a,c) is (A,C), and the highest probability sequences is (B,D) when we tag (c,d).

6 Q 7

Algorithm 2 Maximum-probability-sequence(θ^*,T,V,K)

initialize table D of size $T \times |K|$
D[0][START]=1, and D[0][y]=0 for all other y in K.
for $t = 1 : T - 1$ **do**
  for $y \in K$ **do**
    $D[t][y]; = max_{v \in V, k \in K} D[t-1][v][k] \times \theta(y|k)$
  end for
end for
$D[T][STOP] = max_{k \in K} D[T-1][k] \times \theta^*(STOP|k)$
for $t = 1 : T - 1$ **do**
  $y_t = argmax_y(D[t][y])$
  $x_t = argmax_x(D[t][y_t] \times \vartheta(x|y_t))$
  Store x_t, y_t and the maximum probability $D[t][y_t] \times \vartheta(x|y_t)$.
end for
Return the reconstructed sequences and their probability.

The recursion step involves iterating over T time steps and computing the maximum probability for each tag in K at each time step. Then compute the

maximum over all possible preceding word and tag combinations K*V. So the time complexity of the recursion step is $O(T|K|^2|V|)$.