NLP תרגיל 2

מגישים: ליאור גפן 315635136
יותם כהן 319072864

חלק תיאורטי

1. (10 pts) Consider this (toy) biological setup:
A cell can be in one of two states - H, for high GC-content, and L for low GC. On each time step the cell produces one nucleotide, A,C,T or G, and might also change its state. The probability of changing from state H to L is 0.5, and from state L to H is 0.4.
In state H the probabilities for producing nucleotides are 0.2 for A, 0.3 for C, 0.3 for G and 0.2 for T. In L the probabilities are 0.3 for A, 0.2 for C, 0.2 for G and 0.3 for T.
Consider the nucleotide sequence $S = ACCGTGCA$. Use the Viterbi algorithm to find the best state-sequence and calculate the probability of S given this state-sequence. Assume the previous state before S was H.

1) $\mathbb{P}(H|H) = \frac{1}{2}$, $\mathbb{P}(L|H) = \frac{1}{2}$

$\mathbb{P}(H|L) = 0.4$ $\quad \mathbb{P}(L|L) = 0.6$

$\mathbb{P}(A|H) = 0.2$ $\quad \mathbb{P}(C|H) = 0.3$ $\quad \mathbb{P}(G|H) = 0.2$ $\quad \mathbb{P}(T|H) = 0.2$

$\mathbb{P}(A|L) = 0.3$ $\quad \mathbb{P}(C|L) = 0.2$ $\quad \mathbb{P}(G|L) = 0.2$ $\quad \mathbb{P}(T|L) = 0.3$

	START	A	C	C	G	T	G	C	A
	0	1	2	3	4	5	6	7	8
H	1	0.1	0.018	2.7×10^{-3}	4.05×10^{-4}	4.05×10^{-5}	7.29×10^{-6}	1.0935×10^{-6}	1.0935×10^{-7}
L	0	0.15	0.018	2.16×10^{-3}	2.7×10^{-4}	6.075×10^{-5}	7.29×10^{-6}	8.748×10^{-7}	1.64025×10^{-9}

אכן שהתיוג בעל ההסתברות הגבוהה ביותר לרצף S הינו:

H, L, H, H, H, L, H, H, L

(חישובים למטה) -

$\pi(0,H) = 1$

$\pi(1,H) = P(H|H) \cdot e(A|H) = 0.5 \cdot 0.2 = 0.1$

$\pi(1,L) = P(L|H) \cdot e(A|L) = 0.5 \cdot 0.3 = 0.15$

$\pi(2,H) = \max\{\{\pi(1,H) \cdot P(H|H) \cdot e(C|H)\} \cdot \{\pi(1,L) \cdot P(H|L) \cdot e(C|H)\}\} =$

$\max\{0.1 \cdot 0.5 \cdot 0.3, \; 0.15 \cdot 0.4 \cdot 0.3\} = \max\{0.015, \; 0.018\} = 0.018$

$\pi(2,L) = \max\{\{\pi(1,H) \cdot P(L|H) \cdot e(C|L)\}, \{\pi(1,L) \cdot P(L|L) \cdot e(C|L)\}\} =$

$\max\{0.01, 0.018\} = 0.018$

$\pi(3,H) = 0.018 \cdot e(C|H) \cdot \max\{P(H|H), P(H|L)\} = 0.018 \cdot 0.3 \cdot 0.5 = 2.7 \times 10^{-3}$

$\pi(3,L) = 0.018 \cdot e(C|L) \cdot \max\{P(L|H), P(L|L)\} = 0.018 \cdot 0.2 \cdot 0.6 = 2.16 \times 10^{-3}$

$\pi(4,H) = \max\{\pi(3,H) \cdot P(H|H) \cdot e(G|H), \; \pi(3,L) \cdot P(H|L) \cdot e(G|L)\} =$

$\max\{4.05 \times 10^{-4}, \; 1.728 \times 10^{-4}\} = 4.05 \times 10^{-4}$

$\pi(4,L) = \max\{\pi(3,H) \cdot P(L|H) \cdot e(G|L), \; \pi(3,L) \cdot P(L|L) \cdot e(G|L)\} =$

$\max\{2.7 \times 10^{-4}, \; 2.592 \times 10^{-4}\} = 2.7 \times 10^{-4}$

$\pi(5,H) = \max\{\pi(4,H) \cdot P(H|H) \cdot P(T|H), \; \pi(4,L) \cdot P(H|L) \cdot P(T|H)\} =$

$\max\{4.05 \times 10^{-5}, \; 2.16 \times 10^{-5}\} = 4.05 \times 10^{-5}$

$\pi(5,L) = \max\{\pi(4,H) \cdot P(L|H) \cdot P(T|L), \; \pi(4,L) \cdot P(L|L) \cdot P(T|L)\} =$

$\max\{6.075 \times 10^{-5}, \; 4.86 \times 10^{-5}\} = 6.075 \times 10^{-5}$

$\pi(6,H) = \max\{\pi(5,H) \cdot P(H|H) \cdot P(G|H), \; \pi(5,L) \cdot P(H|L) \cdot P(G|H)\} =$

$\max\{6.075 \times 10^{-6}, \; 7.29 \times 10^{-6}\} = 7.29 \times 10^{-6}$

$$\pi(6,L) = \max\{\pi(5,H)\cdot \mathbb{P}(L|H)\cdot \mathbb{P}(G|L),\ \pi(5,L)\cdot \mathbb{P}(L|L)\cdot \mathbb{P}(G|L)\} =$$

$$\max\{4.05\times10^{-6},\ 7.29\times10^{-6}\} = 7.29\times10^{-6}$$

$$\pi(7,H) = \max\{\pi(6,H)\cdot \mathbb{P}(H|H)\cdot \mathbb{P}(C|H),\ \pi(6,L)\cdot \mathbb{P}(H|L)\cdot \mathbb{P}(C|L)\} =$$

$$\max\{1.0935\times10^{-6},\ 5.832\times10^{-7}\} = 1.0935\times10^{-6}$$

$$\pi(7,L) = \max\{\pi(6,H)\cdot \mathbb{P}(L|H)\cdot \mathbb{P}(C|L),\ \pi(6,L)\cdot \mathbb{P}(L|L)\cdot \mathbb{P}(C|L)\} =$$

$$\max\{7.29\times10^{-7},\ 8.748\times10^{-7}\} = 8.748\times10^{-7}$$

$$\pi(8,H) = \max\{\pi(7,H)\ \mathbb{P}(H|H)\cdot \mathbb{P}(A|H),\ \pi(7,L)\cdot \mathbb{P}(H|L)\cdot \mathbb{P}(A|H)\} =$$

$$1.0935\times10^{-7}$$

$$\pi(8,L) = \max\{\pi(7,H)\cdot \mathbb{P}(L|H)\cdot \mathbb{P}(A|L),\ \pi(7,L)\cdot \mathbb{P}(L|L)\cdot \mathbb{P}(A|L)\}$$

$$1.64025\times10^{-7}$$

2. (10 pts) In class we saw the trigram HMM model and the corresponding Viterbi algorithm. We will now make two main changes. First, we will consider a four-gram tagger, where p takes the form:

$$p(x_1 \cdots x_n, y_1 \cdots y_{n+1}) = \prod_{i=1}^{n+1} q(y_i|y_{i-3}, y_{i-2}, y_{i-1}) \prod_{i=1}^{n} e(x_i|y_i) \tag{1}$$

We assume in this definition that $y_0 = y_{-1} = y_{-2} = *$, where $*$ is the START symbol, $y_{n+1} = STOP$, and $y_i \in \mathcal{K}$ for $i = 1 \cdots n$, where $\mathcal{K}$ is the set of possible tags in the HMM.
Second, we consider a version of the Viterbi algorithm that takes as input **an integer** n (and not a sentence $x_1 \cdots x_n$ as we saw in class) and finds

$$\max_{y_1 \cdots y_{n+1}, x_1 \cdots x_n} p(x_1 \cdots x_n, y_1 \cdots y_{n+1})$$

for a four-gram tagger, as defined in Equation 1. $x_1 \cdots x_n$ may range over the values of some fixed vocabulary $\mathcal{V}$. Complete the following pseudo-code of this version of the Viterbi algorithm for this model . The pseudo-code must be efficient.

Input: An integer n, parameters $q(w|t, u, v)$ and $e(x|s)$.
Definitions: Define $\mathcal{K}$ to be the set of possible tags. Define $\mathcal{K}_{-2} = \mathcal{K}_{-1} = \mathcal{K}_0 = \{*\}$, and $\mathcal{K}_k = \mathcal{K}$ for $k = 1 \cdots n$. Define $\mathcal{V}$ to be the set of possible words.
Initialization: $\cdots$
Algorithm: $\cdots$
Return: $\cdots$

Initialization: $\pi(0, *, *, *) = 1$

Algorithm:

\# For $k = 1 \dots n$:

For $u \in K_{k-2}$, $v \in K_{k-1}$, $w \in K_k$,

$$\pi(k, u, v, w) = \max_{s \in K_{k-3},\, x \in V} \{\pi(k-1, s, u, v) \cdot \mathbb{P}(w | s, u, v) \cdot e(x|w)\}$$

$$bp(k, u, v, w) = \arg\max_{s \in K_{k-3},\, x \in V} \{\pi(k-1, s, u, v) \cdot \mathbb{P}(w | s, u, v) \cdot e(x|w)\}$$

\# set $(y_{n-2}, y_{n-1}, y_n) = \arg\max_{u,v,w} \{\pi(n, u, v, w) \cdot \mathbb{P}(STOP | u, v, w)\}$

\# denote $bp(k, x, y, z)[0] \in K$, $bp(k, x, y, z)[1] \in V$.

\# for $k = 1 \dots n-3$ $\quad y_k = bp(k+3, y_{k+1}, y_{k+2}, y_{k+3})[0]$

$k = 1, \dots, n$ $\quad x_k = bp(k, y_{k-2}, y_{k-1}, y_k)[1]$

\# **Return** $(x_1, \dots x_n, y_1, \dots, y_n)$

Practical Part Results

```
Qb - ii
MLE tag classifier:
The error rate of seen words is: 0.0701023967593114
The error rate of unseen words is: 0.743455497382199
The error rate of all words is: 0.14701485099172729

Qc - iii
Bigram HMM:
The error rate of seen words is: 0.16957353437605494
The error rate of unseen words is: 0.7757417102966842
The error rate of all words is: 0.238811920661816

Qd - iii
Bigram HMM with Add-1 smoothing:
The error rate of seen words is: 0.1436930347698886
The error rate of unseen words is: 0.712914485165794
The error rate of all words is: 0.2087112528655437

Qe - ii
Bigram HMM with Pseudo-words:
The error rate of seen words is: 0.16000900191290646
The error rate of unseen words is: 0.45986038394415363
The error rate of all words is: 0.1942589454799163

Qe - iii
Bigram HMM with Add-1 smoothing and Pseudo-words:
The error rate of seen words is: 0.12703949589287722
The error rate of unseen words is: 0.4432809773123909
The error rate of all words is: 0.1631615668294628
```

```
Confusion matrix investigation:
 for the true POS ', the most frequent POS mistakenly predicted is .
 for the true POS '', the most frequent POS mistakenly predicted is NN
 for the true POS (, the most frequent POS mistakenly predicted is NP
 for the true POS ), the most frequent POS mistakenly predicted is ,
 for the true POS *, the most frequent POS mistakenly predicted is NP
 for the true POS ,, the most frequent POS mistakenly predicted is '
 for the true POS --, the most frequent POS mistakenly predicted is VBD
 for the true POS ., the most frequent POS mistakenly predicted is '
 for the true POS :, the most frequent POS mistakenly predicted is ,
 for the true POS ABN, the most frequent POS mistakenly predicted is ,
 for the true POS ABX, the most frequent POS mistakenly predicted is AT
 for the true POS AP, the most frequent POS mistakenly predicted is NP
 for the true POS AT, the most frequent POS mistakenly predicted is NP
 for the true POS BE, the most frequent POS mistakenly predicted is '
 for the true POS BED, the most frequent POS mistakenly predicted is IN
 for the true POS BEDZ, the most frequent POS mistakenly predicted is '
 for the true POS BEG, the most frequent POS mistakenly predicted is NP
 for the true POS BEN, the most frequent POS mistakenly predicted is '
 for the true POS BER, the most frequent POS mistakenly predicted is NN
 for the true POS BEZ, the most frequent POS mistakenly predicted is '
 for the true POS CC, the most frequent POS mistakenly predicted is CS
 for the true POS CD, the most frequent POS mistakenly predicted is JJ
 for the true POS CS, the most frequent POS mistakenly predicted is IN
 for the true POS DO, the most frequent POS mistakenly predicted is MD
 for the true POS DOD, the most frequent POS mistakenly predicted is TO
 for the true POS DOZ, the most frequent POS mistakenly predicted is TO
 for the true POS DT, the most frequent POS mistakenly predicted is CS
 for the true POS DTI, the most frequent POS mistakenly predicted is AT
 for the true POS DTS, the most frequent POS mistakenly predicted is AT
 for the true POS EX, the most frequent POS mistakenly predicted is NNS
 for the true POS FW, the most frequent POS mistakenly predicted is NN
 for the true POS HV, the most frequent POS mistakenly predicted is VBD
 for the true POS HVD, the most frequent POS mistakenly predicted is CS
 for the true POS HVG, the most frequent POS mistakenly predicted is IN
 for the true POS HVN, the most frequent POS mistakenly predicted is VBN
 for the true POS HVZ, the most frequent POS mistakenly predicted is ,
```

המשך

```
for the true POS HVZ, the most frequent POS mistakenly predicted is ,
for the true POS IN, the most frequent POS mistakenly predicted is TO
for the true POS JJ, the most frequent POS mistakenly predicted is NN
for the true POS JJR, the most frequent POS mistakenly predicted is NN
for the true POS JJS, the most frequent POS mistakenly predicted is NN
for the true POS JJT, the most frequent POS mistakenly predicted is NN
for the true POS MD, the most frequent POS mistakenly predicted is ''
for the true POS NN, the most frequent POS mistakenly predicted is NP
for the true POS NNS, the most frequent POS mistakenly predicted is NN
for the true POS NP, the most frequent POS mistakenly predicted is NN
for the true POS NPS, the most frequent POS mistakenly predicted is NN
for the true POS NR, the most frequent POS mistakenly predicted is JJ
for the true POS OD, the most frequent POS mistakenly predicted is NP
for the true POS PN, the most frequent POS mistakenly predicted is CC
for the true POS PP, the most frequent POS mistakenly predicted is NP
for the true POS PPL, the most frequent POS mistakenly predicted is WPS
for the true POS PPLS, the most frequent POS mistakenly predicted is NNS
for the true POS PPO, the most frequent POS mistakenly predicted is AT
for the true POS PPS, the most frequent POS mistakenly predicted is PPO
for the true POS PPSS, the most frequent POS mistakenly predicted is HVZ
for the true POS QL, the most frequent POS mistakenly predicted is RB
for the true POS QLP, the most frequent POS mistakenly predicted is NN
for the true POS RB, the most frequent POS mistakenly predicted is NN
for the true POS RBR, the most frequent POS mistakenly predicted is NP
for the true POS RP, the most frequent POS mistakenly predicted is IN
for the true POS TO, the most frequent POS mistakenly predicted is IN
for the true POS VB, the most frequent POS mistakenly predicted is NN
for the true POS VBD, the most frequent POS mistakenly predicted is VBN
for the true POS VBG, the most frequent POS mistakenly predicted is NN
for the true POS VBN, the most frequent POS mistakenly predicted is JJ
for the true POS VBZ, the most frequent POS mistakenly predicted is NN
for the true POS WDT, the most frequent POS mistakenly predicted is AT
for the true POS WP, the most frequent POS mistakenly predicted is IN
for the true POS WPS, the most frequent POS mistakenly predicted is CS
for the true POS WQL, the most frequent POS mistakenly predicted is WRB
for the true POS WRB, the most frequent POS mistakenly predicted is QL
for the true POS ``, the most frequent POS mistakenly predicted is '
```