Assignment 2

Anand Kamat
ID# 260773313

Notes:

- **Several code snippets and algorithms were acquired from nltk and its packages.**
- **The Context Free Grammar is provided in a different file.**
- **The code takes command line arguments in the following formats:**
  - **Python decipher.py ‘path’**
  - **Python decipher.py -laplace ‘path’**
  - **Python decipher.py -lm ‘path’**
  - **Python decipher.py -laplace -lm ‘path’**
- **Since the question didn’t specify how the data from Assignment 1 needs to be accessed I have left it as it is. I would request the TA to change the path of the files on Line 166 and Line 170 to load the dataset.**

Question1

Question 1

That that is is
That that is not is not
Is that it
Is it that good.

C N V V
C N V N V N
V N N
V N N J

Initial
~~Emission~~ :

$\pi_C = 2/4$
$\pi_V = 2/4$
$\pi_N = 0$
$\pi_J = 0$

Smoothing: $\pi^*_C = 3/8$
$\pi^*_V = 3/8$
$\pi^*_N = 1/8$
$\pi^*_J = 1/8$

Transition :

$C :\rightarrow \; P(C \rightarrow N) = 1$

Smoothing: $P^*(C \rightarrow N) = 3/6$
$P^*(C \rightarrow C) = 1/6$
$P^*(C \rightarrow V) = 1/6$
$P^*(C \rightarrow J) = 1/6$

$N :\rightarrow \; P(N \rightarrow V) = 3/6$
$P(N \rightarrow N) = 2/6$
$P(N \rightarrow J) = 1/6$

Smoothing: $P^*(N \rightarrow V) = 4/10$
$P^*(N \rightarrow N) = 3/10$
$P^*(N \rightarrow J) = 2/10$
$P^*(N \rightarrow C) = 1/10$

$V :\rightarrow \; P(V \rightarrow V) = 1/5$
$P(V \rightarrow N) = 4/5$

Smoothing: $P^*(V \rightarrow V) = 2/9$
$P^*(V \rightarrow N) = 5/9$
$P^*(V \rightarrow C) = 1/9$
$P^*(V \rightarrow J) = 1/9$

J:→ $P^*(J \to V) = 1/4$

smoothing $P^*(J \to N) = 1/4$

$P^*(J \to C) = 1/4$

$P^*(J \to J) = 1/4$

Emission :

C:) $P(C \to \text{'that'}) = 2/2$

Smoothing : $P^*(C \to \text{'that'}) = 3/7$

$P^*(C \to \text{'is'}) = 1/7$

$P^*(C \to \text{'not'}) = 1/7$

$P^*(C \to \text{'it'}) = 1/7$

$P^*(C \to \text{'good'}) = 1/7$

N:) $P(N \to \text{'that'}) = 4/8$

$P(N \to \text{'not'}) = 2/8$

$P(N \to \text{it}) = 2/8$

π

smoothing: $P^*(N \to \text{that}) = 5/13$

$P^*(N \to \text{is}) = 1/13$

$P^*(N \to \text{'not'}) = 3/13$

$P^*(N \to \text{it}) = 3/13$

$P^*(N \to \text{good}) = 1/13$

V:) $P(V \to \text{is}) = 6/6$

Smoothing : $P^*(V \to \text{that}) = 1/11$

$P^*(V \to \text{is}) = 7/11$

$P^*(V \to \text{not}) = 1/11$

$P^*(V \to \text{it}) = 1/11$

$P^*(V \to \text{good}) = 1/11$

J:) $(J \to \text{good}) = 1/1$

Smoothing : $P^*(J \to \text{that}) = 1/6$

$P^*(J \to \text{is}) = 1/6$

$P^*(J \to \text{not}) = 1/6$

$P^*(J \to \text{it}) = 1/6$

$P^*(J \to \text{good}) = 2/6$

	Not	that	good
C	$\left(3/56\right)$	$\left(\frac{3}{786}\right)$	$\left(\frac{1}{7390}\right)$
N	$\left(3/104\right)$	$\left(5/528\right)$	$\left(1/4576\right)$
V	$\left(3/88\right)$	$\left(\frac{3}{2860}\right)$	$\left(1/2904\right)$
J	$\left(1/56\right)$	$\left(1/672\right)$	$\left(1/1584\right)$

Calculations

I

$$P(V \to Not) = 3/8 \cdot 1/11 = 3/88$$
$$P(C \to Not) = 3/8 \cdot 1/7 = 3/56$$
$$P(N \to Not) = 1/8 \cdot 3/13 = 3/104$$
$$P(J \to Not) = 1/8 \cdot 1/6 = 1/56$$

II

$$P(C \to that) = \max\{(3/7 \cdot 1/6 \cdot 3/56), (3/7 \cdot 1/10 \cdot 3/104), (3/7 \cdot 1/9 \cdot 3/88), (3/7 \cdot 1/4 \cdot 1/56)\}$$
$$= \max\{(3/786), (9/7280), (1/616), (3/1568)\}$$

$$P(N \to that) = \max\{(4/8 \cdot 3/6 \cdot 3/56), (4/8 \cdot 3/10 \cdot 3/104), (4/8 \cdot 5/9 \cdot 3/88), (4/8 \cdot 1/4 \cdot 1/56)\}$$

$$= \max\{(3/448), (9/2080), (5/528), (1/448)\}$$

$$P(V \to that) = \max\{(1/11 \cdot 1/6 \cdot 3/56), (1/11 \cdot 4/10 \cdot 3/104), (1/11 \cdot 2/9 \cdot 3/88), (1/11 \cdot 1/4 \cdot 1/56)\}$$
$$= \max\{(1/1232), (3/2860), (1/1452), (1/2464)\}$$

$$P(J \to that) = \max\{(1/6 \cdot 1/6 \cdot 3/56), (1/6 \cdot 2/10 \cdot 3/104), (1/6 \cdot 1/9 \cdot 3/88), (1/6 \cdot 1/4 \cdot 1/56)\}$$
$$= \max\{(1/1584), (1/1040), (1/672), (1/1344)\}$$

III $P(C \to good) = \max\{(1/7 \cdot 1/6 \cdot 3/786), (1/7 \cdot 1/10 \cdot 5/528), (1/7 \cdot 1/9 \cdot 3/2860), (1/7 \cdot 1/4 \cdot 1/672)\}$

$= \max\{(1/11004), (1/7,392), (1/60,060), (1/18,816)\}$

$P(N \to good) = \max\{(1/13 \cdot 3/62 \cdot 3/786), (1/13 \cdot 3/10 \cdot 5/528), (1/13 \cdot 5/93 \cdot 3/2860), (1/13 \cdot 1/4 \cdot 1/672)\}$

$= \max\{(1/6812), (1/4576), (1/22,308), (1/34944)\}$

$P(V \to good) = \max\{(1/11 \cdot 1/62 \cdot 3/786), (1/11 \cdot 2/5 \cdot 5/528), (1/11 \cdot 2/9 \cdot (3/2860)), (1/11 \cdot 1/4 \cdot 1/672)\}$

$= \max\{(1/17,292), (1/2904), (1/47190), (1/29,568)\}$

$P(J \to good) = \max\{(1/3 \cdot 1/6 \cdot 3/786), (1/3 \cdot 1/8 \cdot 5/528), (1/3 \cdot 1/9 \cdot 3/2860), (1/3 \cdot 1/4 \cdot 1/672)\}$

$= \max\{(1/4716), (1/1584), (1/25740), (1/8064)\}$

	Not	that	good
C	0.05357	3.8168×10^{-3}	1.3532×10^{-4}
N	0.02884	9.4697×10^{-3}	2.1853×10^{-4}
V	0.03409	1.04895×10^{-3}	3.4435×10^{-4}
J	0.01786	1.4881×10^{-3}	6.3131×10^{-4}

Order of Tags:

$C \to N \to J$

Question 2

The advantages of using a CFG over FSA is the use of recursive process to quantify more conditions and states than the FSAs. As we know FSAs cannot represent language of the form {0^n 1^n|n>=1} i.e the set of strings containing one or more zeros followed by an equal number of ones.

The main drawback is that the model lacks context. The POS tags can vary depending on the context at times. CFGs fails when the context affects the language and the tags.

The model I created fails to account for conjugations, concatenations and lacks directive pronouns like ‘mon’, ‘ton’ etc. The model may also fail in some cases when the passive form of the sentence may be used. Another limitation are cases like ‘le Canada’ which the model does not account for.

Question 3:

The following were the observations given by the HMM tagger:

- Without Smoothing
  Cipher 1: 9.05%
  Cipher 2: 14%
  Cipher 3: 22.54%

- With Smoothing
  Cipher 1: 97.69%
  Cipher 2: 83.60 %
  Cipher 3: 22.80%

- With Improved PlainText modelling and without Smoothing
  Cipher 1: 73.44%
  Cipher 2: 74.21%
  Cipher 3: 22.80%

- With Improved PlainText Modelling and Smoothing
  Cipher 1: 95.39%
  Cipher 2: 85.14%
  Cipher 3: 23.31%

The first observation would be the huge spike in performance with smoothing. This is because in absence of smoothing the transition probability distribution matrix is almost full of zeros as it lacks to account for the unknown states and assigns them a value of zero. With laplace smoothing (add-1 smoothing) we see the performance increases tremendously. The first cipher shows the highest growth in performance as the cipher is based on a direct relation. Similar results are seen for the second cipher as well. However the relationship in cipher3 is based on the character three spaces away making the learning more difficult.

The next iteration was the improved plaintext modelling which showed a significant improvement in performance as the transition matrix is updated and now accounts for more data and transitional information. This boosts the performance of the algorithm. However in the case of Cipher1 we see that the performance drops slightly as the pre-processing of the data eliminated several direct relationships (such as '\r' , '\n' etc). Nevertheless the performance overall is obviously significantly improved. And as expected Laplace smoothing increases the performance further.

Cipher1 shows near perfect performance due to its direct relation followed by Cipher2 and Cipher3.