1: Probability Values

NOUN	**FREQ**	**UNSMOOTHED PROB**	**SMOOTHED FREQ**	**SMOOTHED PROB**
maple	600	$\frac{600}{1200} = 0.50$	$\frac{601}{1205} \times 1200 = 598.50$	$\frac{601}{1205} = 0.4988$
oak	400	$\frac{400}{1200} = 0.33$	$\frac{401}{1205} \times 1200 = 399.33$	$\frac{401}{1205} = 0.3328$
pine	180	$\frac{180}{1200} = 0.15$	$\frac{401}{1205} \times 1200 = 180.24$	$\frac{181}{1205} = 0.1502$
spruce	20	$\frac{20}{1200} = 0.01$	$\frac{401}{1205} \times 1200 = 20.91$	$\frac{21}{1205} = 0.0174$
aspen	0	$\frac{0}{1200} = 0.00$	$\frac{401}{1205} \times 1200 = 0.9958$	$\frac{1}{1205} = 0.0008$

2: Recursive Transition Networks

1. **Grammar A and Grammar B:** <u>DIFFERENT</u>
   Grammar A will not parse a sequence starting with a *noun*.

   For example: a Noun Phrase *"noun noun"* can be parsed by *grammar B*, while this cannot be parsed by *grammar A*

2. **Grammar A and Grammar C:** <u>DIFFERENT</u>
   Grammar C will not parse a sequence which will not follow the condition that an adjective comes after an article.

   For example: the sequence *"art noun"* can be parsed by *grammar A*, but it cannot be parsed by *grammar C*.

3. **Grammar A and RTN-2:** <u>DIFFERENT</u>
   Grammar A can only accept a sequence starting with an *article* while *RTN-2* can accept any sequence starting with an *adjective* or an *article*.

   For example: a sequence *"adj noun noun"* can be parsed by *RTN-2* but it cannot be parsed by *Grammer A*.

4. **Grammar A and RTN-3:** <u>DIFFERENT</u>
   Grammar A can accept a sequence starting with an *art* followed by a *noun* while RTN-3 can only accept a sequence starting with an *art* followed by an *adjective*

   For example: a sequence *"art noun"* can be parsed by *Grammer A* but it cannot be parsed by *RTN-3*.

5. **Grammar B and RTN-2:** <u>SAME</u>
   Grammar B and RTN-2 both will accept same input string.

6. **Grammar C and RTN-1:** <u>DIFFERENT</u>
   Grammar C will accept sequences which terminates in *noun* while *RTN-1* can have sequence terminating in *n*oun or *a*djective.

   For example: a sequence *"art adj"* can't be parsed by *Grammer C* but it can be parsed by *RTN-1*.

7. **Grammar C and RTN-3:** <u>SAME</u>
   Grammar C and RTN-3 both will accept same input string .

8. **RTN-1 and RTN-3:** <u>DIFFERENT</u>
   RTN-3 can accept sequences terminating in *noun* only while *RTN-1* can accept sequence ending in an *adjective* also.

   For example: a sequence *" art adj adj"* can be parsed by *RTN-1* but it cannot be parsed by *RTN-3*.

3: Computing Probabilities

1. $P(the) = \frac{5}{34} = 0.147$
2. $P(VERB) = \frac{6}{34} = 0.176$
3. $P(young \mid girl) = \frac{0}{3} = 0$
4. $P(girl \mid young) = \frac{2}{2} = 1$
5. $P(and \mid woman) = \frac{1}{3} = 0.33$
6. $P(thanked \mid young\ girl) = \frac{0}{2} = 0$
7. $P(five \mid gave\ her) = \frac{1}{2} = 0.5$
8. $P(the \mid ART) = \frac{5}{8} = 0.625$
9. $P(cross \mid NOUN) = \frac{0}{9} = 0$
10. $P(thanked \mid VERB) = \frac{2}{6} = 0.33$
11. $P(NUM \mid PRO) = \frac{1}{2} = 0.5$
12. $P(ART \mid VERB) = \frac{4}{6} = 0.66$

4: Viterbi Algorithm

1. P(light=VERB) = P(VERB | ϕ)$\times$ P(light | VERB) = $0.25 \times 0.50 = 0.125$
2. P(light=NOUN) = P(NOUN | ϕ)$\times$ P(light | NOUN) = $0.7 \times 0.60 = 0.42$
3. P(light=ADJ) = P(ADJ | ϕ)$\times$ P(light | ADJ) = $0.2 \times 0.15 = 0.03$
4. P(shows=VERB) = P(shows |VERB) $\times$ max{P(VERB | NOUN) $\times$ P(light| NOUN), P(VERB | VERB) $\times$ P(light | VERB), P(VERB | ADJ) $\times$ P(light | ADJ) = $0.30 \times \max\{0.50 \times 0.42, 0.40 \times 0.125, 0.10 \times 0.3\} = 0.063$
5. P(shows=NOUN) =P(shows | NOUN) $\times$ max{P(NOUN | NOUN) $\times$ P(light| NOUN), P(NOUN | VERB) $\times$ P(light| VERB), P(NOUN | ADJ) $\times$ P(light|ADJ) = $0.40 \times \max\{0.80 \times 0.42, 0.30 \times 0.125, 0.60 \times 0.3\} = 0.1344$
6. P(shows=ADJ) =P(shows | ADJ) $\times$ max{P(ADJ | NOUN) $\times$ P(light |NOUN), P(ADJ | VERB) $\times$ P(light| VERB), P(ADJ | ADJ) $\times$ P(light | ADJ) = $0.10 \times \max\{0.20 \times 0.42, 0.70 \times 0.125, 0.90 \times 0.003\} = 0.00875$

5: Lexical tag - Forward probabilities

1. $P(light/VERB \mid light) = \dfrac{P(light/VERB)}{P(light)} = \dfrac{0.125}{0.575}$
2. $P(light/NOUN \mid light) = \dfrac{P(light/NOUN)}{P(light)} = \dfrac{0.42}{0.575}$
3. $P(light/ADJ \mid light) = \dfrac{P(light/ADJ)}{P(light)} = \dfrac{0.03}{0.575}$
   FOR PART (d),(e),(f) α values are defined as (Calculated in below steps)
   $\alpha_1 = 0.0789$
   $\alpha_2 = 0.1566$
   $\alpha_3 = 0.01985$
4. $P(shows/VERB \mid light \;\; shows) =$
   P(shows |VERB) $\times$ $SUM\{$P(VERB | NOUN) $\times$ P(light| NOUN), P(VERB | VERB) $\times$ P(light | VERB), P(VERB | ADJ) $\times$ P(light | ADJ)
   $= 0.30 \times SUM\{0.50 \times 0.42 + 0.40 \times 0.125 + 0.10 \times 0.3\} = 0.0789$
   Thus $\alpha_1 = 0.0789$

   Now $\alpha_1 /\ (\ \alpha_1 + \alpha_2 + \alpha_3\ ) = (\ 0.0789/0.2553\ )$
5. $P(shows/NOUN \mid light \;\; shows) =$
   P(shows=NOUN) =P(shows | NOUN) $\times$ $SUM\{$P(NOUN | NOUN) $\times$ P(light| NOUN), P(NOUN | VERB) $\times$ P(light| VERB), P(NOUN | ADJ) $\times$ P(light|ADJ)
   $= 0.40 \times SUM\{0.80 \times 0.42 + 0.30 \times 0.125 + 0.60 \times 0.3\} = 0.1566$
   Thus $\alpha_2 = 0.1566$

   Now $\alpha_2 /\ (\ \alpha_1 + \alpha_2 + \alpha_3\ ) = (\ 0.1566/0.2553\ )$
6. $P(shows/ADJ \mid light \;\; shows) =$
   P(shows=ADJ) =P(shows | ADJ) $\times$ $SUM\{$P(ADJ | NOUN) $\times$ P(light |NOUN), P(ADJ | VERB) $\times$ P(light| VERB), P(ADJ | ADJ) $\times$ P(light | ADJ)
   $= 0.10 \times SUM\{0.20 \times 0.42 + 0.70 \times 0.125 + 0.90 \times 0.003\} = 0.01985$
   Thus $\alpha_3 = 0.01985$

   Now $\alpha_3 /\ (\ \alpha_1 + \alpha_2 + \alpha_3\ ) = (\ 0.01985/0.2553\ )$