1. A)

   a((b+ab)a)*(b+ab)

   B)

   a(b+b(a((b+ab)a)*(b+ab+$+(b+ab)a)+a))

2.

a. No Smoothing

- Bigrams model - approximates the probability of a word given all the previous words by the conditional probability of the preceding word.

$$P(w_n \mid w_1^{n-1}) \approx P(w_n \mid w_{n-1})$$

$$P(w_1^n) \approx \prod_{k=1}^{n} P(w_k \mid w_{k-1})$$

Input Sentence: The Fed chairman warned that the board 's decision is bad

P(bad|is) = 0

P(is | decision) = 0

P(decision | board ’s) = 6/17 = 0.352

P(board’s | the) = 10/1430 = 0.06

P (the | that) = 52/257 = 0.20

P(that| warned) = 1/3 = 0.333

P (warned| chairman) = 0

P (chairman | Fed) = 4/19 = 0.210

P (Fed | The) = 2/152 = 0.013

Probability of sentence = 0

b. **Add-one Smoothing**

C_i - nr of counts for type i

$$C_i^* = (C_i + 1)\,\frac{N}{N+V}$$

Why?

$$C_i^* N + C_i^* V = C_i N + N$$

take $\sum_{i=1}^{V}$

$$\sum_{i=1}^{V} C_i^* = \left(\sum_{i=1}^{V} C_i + \sum_{i=1}^{V} 1\right)\frac{N}{N+V}$$

$$N = (N + V)\,\frac{N}{N+V}$$

Add-One Smoothing

$$C_i^* = (C_i + 1)\,\frac{N}{N + V}$$

$$P_i^* = \frac{(C_i + 1)}{N + V}$$

For bigrams:

$$P^*(w_n \mid w_{n-1}) = \frac{C(w_{n-1}w_n) + 1}{C(w_{n-1}) + V}$$

Input Sentence: The Fed chairman warned that the board 's decision is bad

P(bad|is) = (1+ 0)/ (187+5606) = 1.72 x 10^-4

P(is | decision) = (1+0)/ (17+5606) = 1.77 x 10^-4

P(decision | board's) = (6+1)/ (0+5606) = 1.24 x 10^-3

P(board's | the) = (10+1)/(1430 + 5606) = 1.56 x 10^-3

P (the | that) = (52+1)/ (6 + 5606) = 9.44 x 10^-3

P(that| warned) = (1+1)/ (3 + 5606) = 3.56 x 10^-4

P (warned| chairman) = 1 / (554 + 5606) = 1.62 x 10^-4

P (chairman | Fed) = (4+1)/ (19 + 5606) = 8.88 x 10^-4

P (Fed | The) = (2+1)/(152+5606) = 5.21 x 10^-4

Probability of sentence = 1.69 x 10^-30

Good Turing Smoothing

- Nc - the number of N-grams that occur c times.

$$N_c = \sum_{X:count\ (x)=c} 1$$

- Good – Turing smoothing estimates the probability of N-grams that occur c times by the probability of N-grams that occur c + 1 times in the corpus.

$$c^* = (c+1)\frac{N_{c+1}}{N_c}$$

For N_0 –

$$P^*_{GT}\,(things\ with\ frequency\ zero\ in\ training) = \frac{N_1}{N}$$

Bigram	CStar	Probability
('The', 'Fed')	0.9891373801916933	3.47822413739255e-05
('Fed', 'chairman')	2.5298804780876494	8.896126584456184e-05
('chairman', 'warned')	0	0.5397004008720726
('warned', 'that')	0.20393536617148814	7.171227448185109e-06
('that', 'the')	53.0	0.0018637034953231592
('the', "board's")	0	0.5397004008720726
("board's", 'decision')	0	0.5397004008720726
('decision', 'is')	0.20393536617148814	7.171227448185109e-06
('is', 'bad')	0	0.5397004008720726

Probability = 0

3. a.

Question: The_DT standard_?? Turbo_NN engine_NN is_VBZ hard_JJ to_TO work_??

Word	Most Probable Tag
The	DT
standard	NN
Turbo	NN
engine	NN
is	VBZ
hard	JJ
to	TO
work	NN

3 b.

$$\hat{t}_1^n = \underset{t_1^n}{\operatorname{argmax}} \; \overbrace{P(w_1^n|t_1^n)}^{\text{likelihood}} \; \overbrace{P(t_1^n)}^{\text{prior}}$$

Question: The_DT standard_?? Turbo_NN engine_NN is_VBZ hard_JJ to_TO work_??

Ans:

The_DT standard_NN Turbo_NN engine_NN is_VBZ hard_JJ to_TO work_NN

Prior probability of each tag :

tag = nnp prior = 0.10889331800922357

tag = , prior = 0.05595239827167319

tag = rb prior = 0.02285523080727992

tag = jj prior = 0.05634519981960225

tag = nn prior = 0.16151417722623915

tag = vbz prior = 0.019043600971820125

tag = dt prior = 0.09146456784555625

tag = pos prior = 0.008670730465397094

tag = . prior = 0.03290804079316816

tag = nns prior = 0.05401748694298558

tag = in prior = 0.10918428211880064

tag = wp prior = 0.002371357493053232

tag = vbd prior = 0.03318445669726639

tag = vbg prior = 0.01623579731440127

tag = $ prior = 0.00577563757510511

tag = cd prior = 0.034450150573926704

tag = to prior = 0.02256426669770284

tag = vbn prior = 0.02010561997177648

tag = wrb prior = 0.003898919068332921

tag = cc prior = 0.02423731032777107

tag = md prior = 0.012075010547448972

tag = vb prior = 0.02906731454675066

tag = prp$ prior = 0.009238110479072406

tag = jjr prior = 0.0028223518628977115

tag = pdt prior = 0.00027641590409822946

tag = rp prior = 0.002836900068376566

tag = rbr prior = 0.0015566579862373977

tag = wdt prior = 0.005615607314837715

tag = `` prior = 0.0045099436984447966

tag = prp prior = 0.01773426247872325

tag = vbp prior = 0.008932598164016469

tag = '' prior = 0.00427717241078313

tag = : prior = 0.005470125260049173

tag = ex prior = 0.0006546692465484382

tag = nnps prior = 0.001891266712251044

tag = jjs prior = 0.002124037999912711

tag = -lrb- prior = 0.0032297016163056287

tag = -rrb- prior = 0.0033024426436998994

tag = rbs prior = 0.00040734975340791715

tag = wp$ prior = 0.00030551231505593785

Likelihood of a word given tag = number of words with a particular tag/ Total number of tag occurances

Probability of word given tag: = likelihood_of_word_given_tag * prior

p(DT|The):0.053479203340267976

p(JJ|standard):4.364461643656255e-05

p(NN|standard):0.00010183743835197929 - highest

p(NN|Turbo):2.9096410957708368e-05

p(NNP|engine):2.9096410957708368e-05

p(NN|engine):0.00021822308218281275

p(VBZ|is):0.0065030478490478195

p(RB|hard):1.454820547885418e-05

p(JJ|hard):4.364461643656255e-05

p(NNP|to):1.4548205478854184e-05

p(TO|to):0.022549718492223984

p(NN|work):0.000037825334245020876

p(VB|work):0.00010183743835197929

p(VBP|work):0.00010183743835197929 - Highest

Thus, standard is NN and work is VBP