NLP Programming Assignment 3

Name: Priyank Devpura **UIN**: 225002461

Run command: Viterbi.py probs.txt sents.txt

The viterbi.py takes 2 inputs: probs.txt and sents.txt from command line

Results and Analysis:

Analysis:

- The Viterbi algorithm is dependent on the probability provided for the words. The probability provided for the testcase is pretty good.
- From the results we can see that the predicted tags for the sentence are accurate. For eg. Mark bears fish. Mark is noun, bears is a word and fish is a noun. These tags are also predicted from the Viterbi algorithm.
- Some of the probabilities in the Viterbi DP table is zero as for a big sentence, we keep on multiplying with smaller probability values and since it's only up to 10 decimal digits so it turns out to be zero.
- The Best sequence probability is calculated by taking 'fin' into account.

Following are the results:

PROCESSING SENTENCE: bears fish

FINAL VITERBI NETWORK
P(bears = noun)= 0.0160000000
P(bears = verb)= 0.0020000000
P(bears = inf)= 0.0000000100
P(bears = prep)= 0.0000000100
P(fish = noun)= 0.0001232000
P(fish = verb)= 0.0007280000
P(fish = inf)= 0.0000000440
P(fish = prep)= 0.0000004800

FINAL BACKPTR
Backptr(fish = noun)= verb
Backptr(fish = verb)= noun
Backptr(fish = inf)= verb
Backptr(fish = prep)= noun

BEST TAG SEQUENCE HAS PROBABILITY = 0.0003640000
fish -> verb
bears -> noun

FORWARD ALGORITHM RESULTS
P(bears = noun)= 0.0160000000
P(bears = verb)= 0.0020000000
P(bears = inf)= 0.0000000100
P(bears = prep)= 0.0000000100
P(fish = noun)= 0.0001233287
P(fish = verb)= 0.0007280145
P(fish = inf)= 0.0000000442

P(fish = prep)= 0.0000005300

PROCESSING SENTENCE: mark has fish

FINAL VITERBI NETWORK
P(mark = noun)= 0.0720000000
P(mark = verb)= 0.0060000000
P(mark = inf)= 0.0000000100
P(mark = prep)= 0.0000000100
P(has = noun)= 0.0000004620
P(has = verb)= 0.0014040000
P(has = inf)= 0.0000001320
P(has = prep)= 0.0000021600
P(fish = noun)= 0.0000864864
P(fish = verb)= 0.0000000210
P(fish = inf)= 0.0000000309
P(fish = prep)= 0.0000000351

FINAL BACKPTR
Backptr(has = noun)= verb
Backptr(has = verb)= noun
Backptr(has = inf)= verb
Backptr(has = prep)= noun
Backptr(fish = noun)= verb
Backptr(fish = verb)= noun
Backptr(fish = inf)= verb
Backptr(fish = prep)= verb

BEST TAG SEQUENCE HAS PROBABILITY = 0.0000432432
fish -> noun
has -> verb
mark -> noun

FORWARD ALGORITHM RESULTS
P(mark = noun)= 0.0720000000
P(mark = verb)= 0.0060000000
P(mark = inf)= 0.0000000100
P(mark = prep)= 0.0000000100
P(has = noun)= 0.0000004627
P(has = verb)= 0.0014040182
P(has = inf)= 0.0000001327
P(has = prep)= 0.0000023100
P(fish = noun)= 0.0000866446
P(fish = verb)= 0.0000000379
P(fish = inf)= 0.0000000309
P(fish = prep)= 0.0000000351

PROCESSING SENTENCE: mark bears fish

FINAL VITERBI NETWORK
P(mark = noun)= 0.0720000000
P(mark = verb)= 0.0060000000
P(mark = inf)= 0.0000000100
P(mark = prep)= 0.0000000100
P(bears = noun)= 0.0000924000

P(bears = verb)= 0.0009360000
P(bears = inf)= 0.0000001320
P(bears = prep)= 0.0000021600
P(fish = noun)= 0.0000576576
P(fish = verb)= 0.0000042042
P(fish = inf)= 0.0000000206
P(fish = prep)= 0.0000000234

FINAL BACKPTR
Backptr(bears = noun)= verb
Backptr(bears = verb)= noun
Backptr(bears = inf)= verb
Backptr(bears = prep)= noun
Backptr(fish = noun)= verb
Backptr(fish = verb)= noun
Backptr(fish = inf)= verb
Backptr(fish = prep)= verb

BEST TAG SEQUENCE HAS PROBABILITY = 0.0000288288
fish -> noun
bears -> verb
mark -> noun

FORWARD ALGORITHM RESULTS
P(mark = noun)= 0.0720000000
P(mark = verb)= 0.0060000000
P(mark = inf)= 0.0000000100
P(mark = prep)= 0.0000000100
P(bears = noun)= 0.0000925442
P(bears = verb)= 0.0009360122
P(bears = inf)= 0.0000001327
P(bears = prep)= 0.0000023100
P(fish = noun)= 0.0000578162
P(fish = verb)= 0.0000042243
P(fish = inf)= 0.0000000206
P(fish = prep)= 0.0000000262

PROCESSING SENTENCE**: mark likes to fish for fish**

FINAL VITERBI NETWORK
P(mark = noun)= 0.0720000000
P(mark = verb)= 0.0060000000
P(mark = inf)= 0.0000000100
P(mark = prep)= 0.0000000100
P(likes = noun)= 0.0000004620
P(likes = verb)= 0.0000046800
P(likes = inf)= 0.0000001320
P(likes = prep)= 0.0000021600
P(to = noun)= 0.0000000004
P(to = verb)= 0.0000000000
P(to = inf)= 0.0000010193
P(to = prep)= 0.0000003861
P(fish = noun)= 0.0000000263
P(fish = verb)= 0.0000000535
P(fish = inf)= 0.0000000000

P(fish = prep)= 0.0000000000
P(for = noun)= 0.0000000000
P(for = verb)= 0.0000000000
P(for = inf)= 0.0000000000
P(for = prep)= 0.0000000031
P(fish = noun)= 0.0000000002
P(fish = verb)= 0.0000000000
P(fish = inf)= 0.0000000000
P(fish = prep)= 0.0000000000

FINAL BACKPTR
Backptr(likes = noun)= verb
Backptr(likes = verb)= noun
Backptr(likes = inf)= verb
Backptr(likes = prep)= noun
Backptr(to = noun)= verb
Backptr(to = verb)= noun
Backptr(to = inf)= verb
Backptr(to = prep)= verb
Backptr(fish = noun)= prep
Backptr(fish = verb)= inf
Backptr(fish = inf)= inf
Backptr(fish = prep)= noun
Backptr(for = noun)= verb
Backptr(for = verb)= noun
Backptr(for = inf)= verb
Backptr(for = prep)= verb
Backptr(fish = noun)= prep
Backptr(fish = verb)= noun
Backptr(fish = inf)= verb
Backptr(fish = prep)= noun

BEST TAG SEQUENCE HAS PROBABILITY = 0.0000000001
fish -> noun
for -> prep
fish -> verb
to -> inf
likes -> verb
mark -> noun

FORWARD ALGORITHM RESULTS
P(mark = noun)= 0.0720000000
P(mark = verb)= 0.0060000000
P(mark = inf)= 0.0000000100
P(mark = prep)= 0.0000000100
P(likes = noun)= 0.0000004627
P(likes = verb)= 0.0000046801
P(likes = inf)= 0.0000001327
P(likes = prep)= 0.0000023100
P(to = noun)= 0.0000000006
P(to = verb)= 0.0000000000
P(to = inf)= 0.0000010196
P(to = prep)= 0.0000004320
P(fish = noun)= 0.0000000294
P(fish = verb)= 0.0000000536
P(fish = inf)= 0.0000000000
P(fish = prep)= 0.0000000000

P(for = noun)= 0.0000000000
P(for = verb)= 0.0000000000
P(for = inf)= 0.0000000000
P(for = prep)= 0.0000000051
P(fish = noun)= 0.0000000003
P(fish = verb)= 0.0000000000
P(fish = inf)= 0.0000000000
P(fish = prep)= 0.0000000000

Limitations:

- It is dependent on the probability of the words with tags. If these are not accurate, then Viterbi will give incorrect results.