*With this code I will be predicting a sequence from an image using three methods:*
*Using the test image data, train image data and train text data I am trying to calculate three probabilities here:*
*A. Initial Probability:*
    *Initial Probability for letter[i] is calculated as the number of statement starting from letter[i] over total number of statements in file.*
*B. Transition Probability:*
    *Transition probability is calculated from the text training file by counting the transitions of letter[i] to letter[j] divided by the total transitions of letter[i]*
    *this gives me P(l[j]|l[i]) where j represents the next alphabet in unobserved sequence and i is the current alphabet*
*C. Emission Probability E:*
    *This E represents the probability of l(i) = letters[j] given O(i) = test_letters[r]*
    *where l(i) is the i-th element of sequence to be predicted, O(i) is the observed variable which is currently set to the r-th pixel arrangement of test_letters*

*Using these three probabilities the prediction is done in three ways:*
*I. Simplified method:*
*In this method,*
*1. I will be calculating the probability by taking each observed alphabet O from test sequence*
*2. Then Each pixel of this observed alphabet O is matched with each of the train alphabets in 'letters'*
*3. Based on number of pixels matched, I am calculating the emission probability E for each alphabet in letters.*
*4. The maximum value of this emission probability E gives the most probable alphabet*
*5. This sequence repeated for each alphabet gives the most probable sequence.*

*II. HMM using variable elimination:*
*In this method,*
*1. let the sequence to be predicted be l1,l2,l3,...,ln*
*2. The observed pixels states be O1,O2,O3...,On*
*3. Here the probability for l1 is found for all letters in LETTERS*
*4. This is done by multiplying the initialProb(l1) by emission_prob(l1|O1)*
*5. Then l1 is eliminated by marginalising all l1 terms over all values of l1*
*6. l1 terms include P(l1)∗P(l2|O2)∗P(l2|l1). This can be replaced by alpha(l2)*
*7. Similarly eliminating l2 terms and l3,l4,...ln terms will give us a lookup table with l1,l2,...ln values*
*8. Selecting the maximum probability values gives the maximum probable sequence*

*III. HMM using Viterbi Algorithm:*
*In this method,*
*1. let the sequence to be predicted be l1,l2,l3,...,ln*
*2. The observed pixels states be O1,O2,O3...,On*
*3. Here the probability for l1 is found for all letters in LETTERS*
*4. This is done by multiplying the initialProb(l1) by emission_prob(l1|O1)*
*5. The max probability from l1 values is taken as the initial probability for predicting l2 value*
*6. probability of l2 is calculated as,*
    *P(l2) = max(initial_prob(l1)∗P(l2|l1)∗P(l2|O2))*
*7. This max value of P(l2) is used as initial probability to calculate P(l3) and so on.*
*8. At each step only the maximum probability value moves forward to the next step.*
*9 At each step the maximum prob alphabet is the predicted alphabet of that test sequence step.*
*10. Hence at the end we get the most probable sequence.*
*11. This process takes less time than variable elimination*

These are the performances I observed:

```
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png
bc.train test-0-0.png
Viterbi Accuracy: 94.1176470588 %
VE Accuracy: 94.1176470588 %
Simple: SUPREME COURT OF THF UN1TED STATES
HMM VE: SUPREME COURT OF THF UN1TED STATES
HMM Viterbi: SUPREME COURT OF THF UN1TED STATES
 (myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png
bc.train test-1-0.png
Viterbi Accuracy: 15.7142857143 %
VE Accuracy: 15.7142857143 %
Simple:                                                1
HMM VE:                                                1
HMM Viterbi:                                                1
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png
bc.train test-2-0.png
Viterbi Accuracy: 86.2068965517 %
VE Accuracy: 86.2068965517 %
Simple: Nos. 14-556. -rguec -pr11 28, 2015 - Dec1cec June 26, 2015
HMM VE: Nos. 14-556.  rguec  pr11 28, 2015 - Dec1cec June 26, 2015
HMM Viterbi: Nos. 14-556.  rguec  pr11 28, 2015 - Dec1cec June 26, 2015
 (myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png
bc.train test-3-0.png
Viterbi Accuracy: 95.0819672131 %
VE Accuracy: 95.0819672131 %
Simple: Together w1th No. 14-562, Tanco et a1. v. Has1am, Governor of
HMM VE: Together w1th No. 14-562, Tanco et a1. v. Has1am, Governor of
HMM Viterbi: Together w1th No. 14-562, Tanco et a1. v. Has1am, Governor of
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png
bc.train test-4-0.png
Viterbi Accuracy: 42.8571428571 %
VE Accuracy: 42.8571428571 %
Simple: . n -ss  ,     1..  1so c     t cr  1 to       me  c r .
HMM VE: . n  ss  ,     1..  1so c     t cr  1 to       me  c r .
HMM Viterbi: . n  ss  ,      ..  1so c     t cr  1 to       me  c r .
 (myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png
bc.train test-5-0.png
Viterbi Accuracy: 95.0 %
VE Accuracy: 95.0 %
Simple: Opinion of the Ccurt
HMM VE: Opinion of the Ccurt
HMM Viterbi: Opinion of the Ccurt
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png
bc.train test-6-0.png
Viterbi Accuracy: 19.0476190476 %
VE Accuracy: 22.2222222222 %
Simple:      m       -  -      -              -  c   n  r   . m   1 g
HMM VE:      m       -                         c   n  r   . m   1 g
HMM Viterbi:      m       -                          c   n      . m   1
 (myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png
bc.train test-7-0.png
Viterbi Accuracy: 91.6666666667 %
VE Accuracy: 89.5833333333 %
Simple: emcoc1es a 1ove that may endure even past ceath.
HMM VE: emcoc1es a 1ove that may endure even past ceath.
HMM Viterbi: emboc1es a 1ove that may endure even past ceath.
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png
bc.train test-8-0.png
Viterbi Accuracy: 15.3846153846 %
VE Accuracy: 15.3846153846 %
Simple:       1
HMM VE:       1
HMM Viterbi:
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png
bc.train test-9-0.png
Viterbi Accuracy: 23.8095238095 %
```

**VE Accuracy: 23.8095238095 %**
Simple:  -  c  m  1  .
HMM VE:  c  m  1  .
HMM Viterbi:  c  m  .
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png bc.train test-10-0.png
**Viterbi Accuracy: 92.1875 %**
**VE Accuracy: 90.625 %**
Simple: The1r p1ea 1s that they do respect 1t, respect 1t so deep1y that
HMM VE: The1r p1ea 1s that they do respect 1t, respect 1t so deep1y that
HMM Viterbi: The1r p1ea is that they co respect it, respect it so ceep1y that
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png bc.train test-11-0.png
**Viterbi Accuracy: 18.3673469388 %**
**VE Accuracy: 18.3673469388 %**
Simple:  -  -  1  1  1 111m  1  .
HMM VE:  1  1  1 111m  1  .
HMM Viterbi:  1  1  1 111m  1  .
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png bc.train test-12-0.png
**Viterbi Accuracy: 82.1428571429 %**
**VE Accuracy: 80.3571428571 %**
Simple: 1he1r hope 1s not to be ccndemned tc 11v  n 1one1iness,
HMM VE: 1he1r hope 1s not to be ccndemned tc 11v  n 1one1iness,
HMM Viterbi: 1he1r hope is not to be ccndemned tc 11v  n 1one1iness,
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png bc.train test-13-0.png
**Viterbi Accuracy: 94.6428571429 %**
**VE Accuracy: 94.6428571429 %**
Simple: exc1uded from one of civi1ization's o1dest institutions.
HMM VE: exc1uded from one of civi1ization's o1dest institutions.
HMM Viterbi: exc1uded from one of civi1ization's o1dest institutions.
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png bc.train test-14-0.png
**Viterbi Accuracy: 78.0 %**
**VE Accuracy: 80.0 %**
Simple: T ey ask for equa1 c1gn1 y 1n  he eyes of  he 1aw.
HMM VE: T ey ask for equa1 c1gn1 y 1n  he eyes of  he 1aw.
HMM Viterbi: T ey ask for equa1 c1gn1 y 1n  e eyes of  he 1aw.
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png bc.train test-15-0.png
**Viterbi Accuracy: 97.5 %**
**VE Accuracy: 95.0 %**
Simple: The Constitut1on grants them that r1ght.
HMM VE: The Constitut1on grants them that r1ght.
HMM Viterbi: The Constitution grants them that r1ght.
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png bc.train test-16-0.png
**Viterbi Accuracy: 18.0555555556 %**
**VE Accuracy: 18.0555555556 %**
Simple:  c  r  1
HMM VE:  c  r  1
HMM Viterbi:  c  r  1
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png bc.train test-17-0.png
**Viterbi Accuracy: 94.1176470588 %**
**VE Accuracy: 88.2352941176 %**

```
Simple: 1t 1s so ordered.
HMM VE: 1t 1s so ordered.
HMM Viterbi: 1t is so ordered.
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png
bc.train test-18-0.png
Viterbi Accuracy: 17.5438596491 %
VE Accuracy: 17.5438596491 %
Simple:          .                                      1
HMM VE:          .                                      1
HMM Viterbi:          .
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$ python ocr.py courier-train.png
bc.train test-19-0.png
Viterbi Accuracy: 26.9230769231 %
VE Accuracy: 28.8461538462 %
Simple:  1N D     B     .   1      ,  nc      ,   ..    n c.
HMM VE:  1N D     B     .   1      ,  nc     N,   ..    n c.
HMM Viterbi:  1N       B     .   1      ,  nc      ,   ..    n c.
(myenv) VINITAs-MacBook-Air:part2 vinitaboolchandani$
```

So the mean performance I got was : **60%**