

2 Viterbi coding

2.1 Pseudocode for Viterbi algorithm

The pseudocode for the Viterbi algorithm is as follows:

'L' is the number of labels

'N' is the number of tokens of a sentence

'start' is the set of start probabilities, in the same order of the labels

'end' is the set of end probabilities, in the same order of the labels

'table' be the matrix to reserve the max probability value of tag sequence

'backpointer' be the matrix to reserve the index of back pointers

// Initialization: the start probabilities of first column

for i (0..L-1):

 table[i][0] = start[i] + emission[0][i]

// Recursion: all other columns, calculate the best values

for j (1, N-1):

 for i (0, L-1):

 s = -(MAX_INT-1)

 for k (0, L-1):

 temp = list((table[k][j - 1] + trans[k][i]))

 s = emission[j][i] + table[k][j - 1] + trans[k][i]

 if s > temp

 table[i][j] = s

 backpointer[i][j] = k

// Last Column: add the end probabilities of last column

for i (0, L-1):

 end = table[i][N - 1] + end[i]

last_seq = np.argmax(end)

score = max(end_prob)

// Construct best sequence, initial s and index minor than 0.

s = -sys.maxint - 1

```
index = -1

//find the best sequence
for i (0, L-1):
    if table[i][-1] > s:
        s = table[i][-1]
        index = i
while(l = N-1; i != -1; i--):
    sequence = [index] + y
    index = back[index][i]
return score, sequence
```