# HW02 Project Report

## 3. Maximum Entropy Tagging

In maximum entropy models, the sequence tagging is done as a multi-label classification task either by doing forward or backward traversal of the given input sequence.

In Ratnaparkhi’s MaxEnt tagger, the forward procedure for tagging the sequence does not always find the best possible sequence of tags because

1. The features for each word in the seq. is the context of 2 words to the right and 2 words to the left and tags of previous 2 words. This means it fails to consider the long distance relationships (outside the context) which might have an influence.
2. As the model depends on previous 2 tags (ti-1, ti-2) if any of themis tagged wrongly then the model might tag the words that comes next incorrectly.

At the same time, backward procedure also does not yield perfect result always. Moreover, it is difficult to combine the information from forward and backward procedure to tag the word. In order to overcome this issues, we need a model that maximizes the likelihood of joint tagging for a given input sequence. Probabilistic Sequence models like Conditional Random Fields, Hidden Markov Model with Viterbi algorithm can be used to collectively determine the most global assignment of tags for the given words in the sequence.

# 5. POS Tag Entropy

The brown corpus is a collection of 500 American English documents spread across various categories whereas Treebank corpus has sentences from Wall Street Journals. The words in these corpora are tagged with parts of speech. The tagged corpora is preprocessed (say Xpre) by converting the words into lower case and there by ignoring the words that appear less than 5 times. The tag entropies for all the words in the corpora are computed.

## 5.1 Top 10 Highest Entropy Words

The words are sorted based upon their tags entropies and the top 10 words are reported below with the format as Word Count\_of\_the\_word Tags Entropy

### 5.1.1 Brown Corpus

northeast 16 {('northeast', 'NR-TL-HL'), ('northeast', 'NN-TL'), ('northeast', 'NR-TL'), ('northeast', 'NR'), ('northeast', 'JJ'), ('northeast', 'JJ-TL')} 2.452819531114783  
chase 18 {('chase', 'NN-TL'), ('chase', 'NN-HL'), ('chase', 'VB'), ('chase', 'NP'), ('chase', 'NP-TL'), ('chase', 'NN')} 2.3718956694089632  
round 75 {('round', 'NN-TL'), ('round', 'NN'), ('round', 'RB'), ('round', 'VB'), ('round', 'IN'), ('round', 'JJ'), ('round', 'JJ-TL'), ('round', 'JJ-HL')} 2.345722433710929  
b 105 {('b', 'NN-HL'), ('b', 'NN-TL-HL'), ('b', 'NP-HL'), ('b', 'NP'), ('b', 'NP-TL'), ('b', 'NN'), ('b', 'NN-TL')} 2.2787839998066235  
right 613 {('right', 'JJ'), ('right', 'NN-HL'), ('right', 'NR'), ('right', 'QL'), ('right', 'RB'), ('right', 'NN'), ('right', 'NN-TL'), ('right', 'JJ-HL'), ('right', 'JJ-TL')} 2.245358265403839  
northwest 25 {('northwest', 'NR'), ('northwest', 'NN-TL'), ('northwest', 'JJ-TL'), ('northwest', 'NR-TL'), ('northwest', 'JJ')} 2.197405811425518  
pro 16 {('pro', 'NN'), ('pro', 'FW-IN-TL'), ('pro', 'IN-HL'), ('pro', 'JJ'), ('pro', 'IN')} 2.149397470347699  
beat 68 {('beat', 'NNS'), ('beat', 'NN-TL-HL'), ('beat', 'VB'), ('beat', 'VBD'), ('beat', 'VBN'), ('beat', 'JJ-TL'), ('beat', 'JJ'), ('beat', 'NN')} 2.0998469626866143  
gross 66 {('gross', 'NN'), ('gross', 'JJ-HL'), ('gross', 'JJ'), ('gross', 'JJ-TL'), ('gross', 'NP-TL'), ('gross', 'NP')} 2.0573358641495325  
rival 12 {('rival', 'NN'), ('rival', 'NN-HL'), ('rival', 'VB'), ('rival', 'JJ'), ('rival', 'JJ-TL')} 2.054585169337799

### 5.1.2 Penn Treebank

hit 10 {('hit', 'VBP'), ('hit', 'VBD'), ('hit', 'VBN'), ('hit', 'NN'), ('hit', 'VB')} 2.170950594454669  
set 29 {('set', 'VBP'), ('set', 'VBD'), ('set', 'VBN'), ('set', 'NN'), ('set', 'VB')} 2.0719814254906335  
close 28 {('close', 'NN'), ('close', 'JJ'), ('close', 'VB'), ('close', 'RB')} 1.8409745639875459  
savings 24 {('savings', 'NNPS'), ('savings', 'NNS'), ('savings', 'NN'), ('savings', 'NNP')} 1.8337993233858494  
lead 20 {('lead', 'VB'), ('lead', 'JJ'), ('lead', 'NN'), ('lead', 'VBP')} 1.765957320949175  
limited 10 {('limited', 'VBN'), ('limited', 'JJ'), ('limited', 'NNP'), ('limited', 'VBD')} 1.7609640474436812  
down 57 {('down', 'NN'), ('down', 'IN'), ('down', 'RP'), ('down', 'NNP'), ('down', 'RB')} 1.7605902374234237  
run 18 {('run', 'VB'), ('run', 'VBN'), ('run', 'NN'), ('run', 'VBP')} 1.7527152789797045  
call 8 {('call', 'VB'), ('call', 'VBP'), ('call', 'NN'), ('call', 'NNP')} 1.75  
cut 28 {('cut', 'VBD'), ('cut', 'VBN'), ('cut', 'NN'), ('cut', 'VB')} 1.7449047091342316

## 5.2 Discussion

Entropy is a measure of uncertainty. As the tag entropy of a word is high, the more uncertain it s in getting the right tag. So, the words that have a wide range of possible tags will have high entropy and is difficult to tag right always. The same trend has been observed in the both the corpora as shown above.

Apart from this, for each corpus the lists of words based upon the no of different tags that the word takes are computed. The counts for the words that take more than 1 tag are shown in Figure 1 for Brown and Penn Treebank. Its observed that, the total no of words having ambiguous tags are 7224 and 1005 for brown and tree bank respectively. Out of them, 68.3% and 77.2% of the words are can possibly have 2 tags for brown and Penn Treebank respectively. So, its easy to disambiguate the tags for these words because the various tags associated with a word are not equally likely. Also, As the number of possible tags that a word takes increases, the overall count of those words decreases.



A: No of possible tags a word can have

(Horizontal Axis)

B: No of words having possible tags count as A

(Vertical Axis)

Figure Brown Corpus and Penn Treebank

Figure 2 Brown Corpus

# 6. HMM POS Tagging

## 6.1 Evaluating multiple models for HMM POS

The Hidden Markov Model is tested with MLE, Laplace, Witten Bell, Simple Good Turing smoothing estimators. Along it with it, a base line system Most likely tag Tagger is used to compare it with HMM’s. The accuracies of these models are reported in Figure 3 when a) all words b) out-of-vocabulary words c) words with out punctuation’s d) top 10 entropy words are used. From Figure 3, clearly smoothing has a key role to play in HMM training. When all words in the test data are considered, there is an absolute increase of 50% from MLE to Laplace. Similar trend has been observed in b) c) and d) cases.

Figure Evaluating multiple HMM's

## 6.3 Learning Curve

The best HMM model is when Witten Bill smoothing is used. The learning curve is plotted in Figure 3 by using 10% to 100% of the training data in the increments of 10 and 100% test data on Penn Treebank. It is observed that as the training data increases, the performance on test data increases. There are many parameters like transition probabilities, symbol emission probabilities and start state probabilities that the model has to learn. In order to have good estimates of these values, the model needs to have good amount of training data. So, we can see there is a jump in accuracy from 68% to 78% when there is an increase of 10% of the training data set from 10% to 20%.



Figure Learning Curve