Assignment-Discussion

HMM Based POS tagging
Using Viterbi Algorithm

Jeet Mehta , 200010036
Anshuman Verma, 200110015
Shatayu Ganvir , 200110103

04/09/2022

Per POS performance

^ : Precision score: 0.9999808061420346, Recall : 1.0, F1 score: 0.99999040261049

DET : Precision score: 0.8990808694611703, Recall : 0.8585089163284738, F1 score: 0.8782683305402278

NOUN : Precision score: 0.8741508746530229, Recall : 0.9304784774976798, F1 score: 0.9013568937203381

ADJ : Precision score: 0.9191036358247201, Recall : 0.8500535794432571, F1 score: 0.8832044473114674

VERB : Precision score: 0.9920402498384856, Recall : 0.9930072385935655, F1 score: 0.9925203918798182

ADP : Precision score: 0.9335678966537125, Recall : 0.9859304180372078, F1 score: 0.959024258434499

PUNCT : Precision score: 0.9430104549167002, Recall : 0.9022310427957146, F1 score: 0.9221416902652466

. : Precision score: 0.9464801345453477, Recall : 0.9121128413212402, F1 score: 0.9274278912562245

ADV : Precision score: 0.9872388797874005, Recall : 0.939452309668097, F1 score: 0.9627417868364423

CONJ : Precision score: 0.7199666042427122, Recall : 0.7828364051425976, F1 score: 0.7498770969904746

PRT : Precision score: 0.9711265647722034, Recall : 0.99995002372355, F1 score: 0.9851813483288318

PRON : Precision score: 0.9497473107551586, Recall : 0.9468606292829056, F1 score: 0.9482821686693613

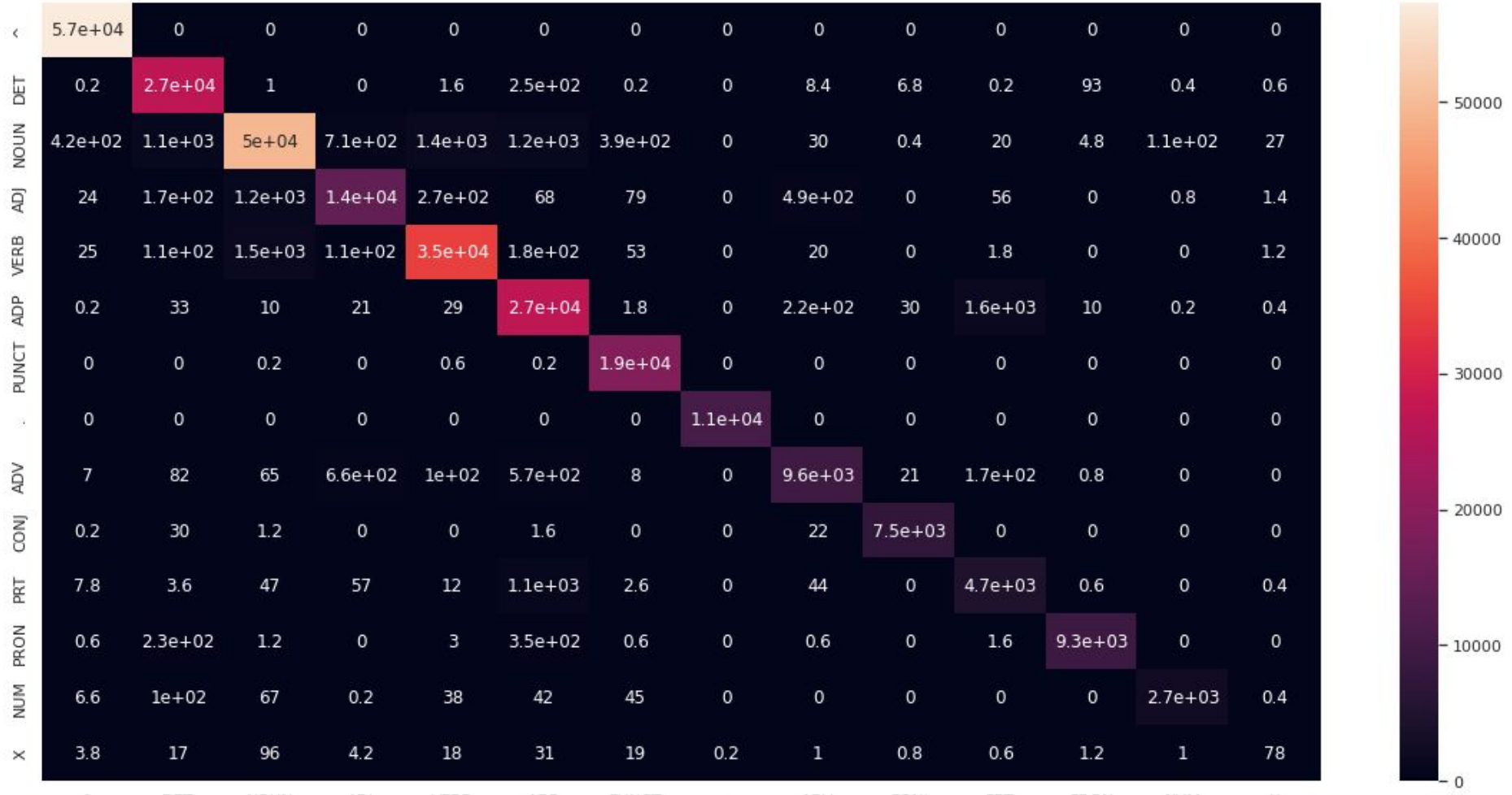
NUM : Precision score: 0.702516038950685, Recall : 0.27305206626236367, F1 score: 0.38942814387931823

X : Precision score: 0.9925365574436004, Recall : 1.0, F1 score: 0.9962540355363124

Problem Statement

- Given a sequence of words, produce the POS tag sequence
- Technique to be used: HMM-Viterbi
- Use Universal Tag Set (12 in number)
- 5-fold cross validation
- Tags:-['^','DET','NOUN','ADJ','VERB','ADP','PUNCT','.','ADV','CONJ','PRT','PRON','NUM','X']

Confusion Matrix (12 X 12) (can give heat map)



Overall performance

- Precision : 0.9164676341419253
- Recall : 0.883890996292618 1
- F-score (3 values)
  - F1-score : 0.892549920447075
  - F0.5-score : 0.9032655908627066
  - F2-score : 0.8865115420607627

Interpretation of confusion (error analysis)

- Through Confusion Matrix we can predict the overall accuracy of our POS Tagger Model.
- The tag ADP has been confused with PRT the most with 1.6e + 03 wrong predictions
-

Data Processing Info (Pre-processing)

Data obtained from brown corpus was cleaned first, lower casing of all the words in all the sentences was done, and then punctuations were handled separately, and word and sentences were tokenized.

The brown corpus with the universal tagset classified all punctuations with the tag '.' and hence to avoid ambiguity with full stop, we replaced '.' for punctuations with 'PUNCT'

To get lexical probabilities we have taken word frequency for the given tag and divided it by all the words appearing for the given tag, to get transition probabilities we have taken frequency of the tag appearing after a specific tag and divided by all possible tags that can appear after the tag.

Inferencing/Decoding Info

As there are multiple tags possible for the given word, decoding the best possible path will take a lot of time and perhaps it may give some error in the end. We need some efficient algorithm to solve this problem

Using Viterbi Algorithm, we will eliminate all those paths that surely won't produce highest probability, and only take those paths that can produce the winning sequence.Given an input of words, we will generate sequence of tags that will be having highest probability as the output.

Marking Scheme

- 1. Demo working- 10/10 (if not, 0)
- 2. Implemented Viterbi and Clarity on Viterbi- 5/5
- 3. Transition and Lexical tables clearly described- 5/5
- 4. Confusion matrix drawn and error analysed- 5/5
- 5. Overall F-score > 90- 10/10, >80 & <=90- 8/10, else 6/10
- 6. Unknown word handling- done (5/5; else 0)

Any thoughts on generative vs. discriminative POS tagging

Generative POS tagging models , like HMM based POS tagging model does not take care of the free order and agglutination(morphemes are heavily joined together),

Also when they there are too many morphemes in the word , then HMM based pos tagging may produce inaccurate output.

In discriminative POS tagging , instead of just POS tag of the previous word, we use feature vectors and then use this feature vectors to get the probability of all the tags. We use maximum entropy markov model(MEMM) in this case to get probability of a tag sequence.

Learning Experience

Data obtained from brown corpus was cleaned first, to get lexical probabilities we have taken word frequency for the given tag and divided it by all the words appearing for the given tag, to get transition probabilities we have taken frequency of the tag appearing after a specific tag and divided by all possible tags that can appear after the tag

HMM - For producing the best possible, tag sequence given a word sequence we are using HMM based POS tagging, transition and lexical probabilities were extracted from brown corpus. According to Markov Assumption, next tag depends only on previous tag. Now, since there can be many tags possible for the given word, therefore for a word sequence, it will be very difficult to predict the best tag sequence if we just consider all the possible tags for a given word using brute force method.

To make the task simpler, We will be using Viterbi Algorithm, i.e, we will eliminate all the paths that surely won't produce highest probability, and only take those paths that can produce the winning sequence.