PART OF SPEECH TAGGING

Vikas Rajpoot
CSE
IIIT Sri City
Andhra Pradesh, India
Email: vikas.r@iiits.in

Dr. Himangshu Sarma
CSE
IIIT Sri City
Andhra Pradesh, India
Email: himangshu.sarma@iiits.in

***Abstract*—In this project I create the part of speech tagger using the basic probabilistic and hidden markov model. probabilistic model is based on the calculate the N-Gram probability for tags and probability that the tag is attach to the word. In hidden markov model I used viterbi algorithm for tag the sentence. for the implementation I use the penn tree bank 10% data which is available at the nltk library. whole dataset(penn tree bank) is not available for free so i use only 10% of that.**

1. Introduction

Tagging is the process of tag the each word in sentence corresponding to a particular part of speech tag, based on its definition and context.

Part of speech tagging is very important in the Natural language processing. In this project i make POS tagger for the English language. I used the Probabilistic model and Hidden markov model.

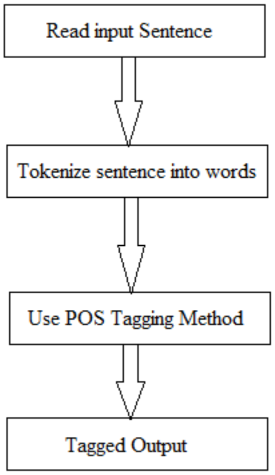

Fig. 1. Part of speech flow chart

An Hidden morkov model is a probabilistic sequence model given a sequence of units. It calculate the probability distribution over possible sequence of labels and chooses the best sequence.

1.1. Motivation

Part of speech tags gives information about words in language. Tag to the word also gives info about the word and its neighbors. Part of speech tagging have application in various tasks such as Information retrieval, parsing, Text to Speech, semantics analysis, language translation and many more. There is almost all the application of NLP required Part of speech tagging as the sub task.

2. State of the art/Background

2.1. TnT – A Statistical Part-of-Speech Tagger [1]

Trigrams'n'Tags (TnT) is an efficient statistical part-of-speech tagger based on the Hidden markov model. Hidden markov models performs as well as the other current approaches, including the maximum entropy framework.
They use the

Model performance	
Penn Tree Bank dataset	Accuracy
known words unknown words overall	97.0 % 85.5 % 96.7 %

Table 1: Accuracy for the TnT model

2.2. SVMTool: A general POS tagger generator based on Support Vector Machines [2]

The SVMtool is a simple part-of-speech tagger based on Support Vector Machines.
They used the fellowing feature for the svm training and testing. word feature, POS feature, ambiguity classes, word bigrams, POS bigrams, word trigrams, POS trigrams, sentence_info, prefixes, suffixes, binary word features, word length.

Model performance and comparison		
	TnT	SVMTools
known acc.	97.0 %	98.08 %
unknown acc.	95.5 %	88.28 %
overall acc.	96.7 %	96.89 %

Table 2: Comparision of the SVMTools with the TnT

2.3. Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms [3]

In this paper they introduce the new algorithm for training tagging models, as alternative to maximum-entropy models or conditional random fields (CRFs). The algorithm rely on Viterbi decoding of training examples, combined with simple additive updates. Maximum-entropy (ME) models are justifiably a very popular choice for tagging problems in Natural Language Processing. Features used in this algorithm are these : Current word, Previous word, Word two back Next word, Word two ahead, Bigram features, Current tag, Previous tag, tag two back, next tag, tag two ahead, Bigram tag features, Trigram tag features. they get the error of 3.28% for the POS tagging.

2.4. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF [4]

The traditional high performance model for POS tagging are linear models, including Hidden Markov Models (HMM) and Conditional Random Fields. In this paper author introduce the Neural network architecture that benefits from both word and character level representations automatically, by using combination of bidirectional LSTM, CNN and CRF. They test this model on Penn Treebank WSJ corpus for part-of-speech (POS) tagging. They obtain 97.55% accuracy for POS tagging.

Dataset Details		
	WSJ	**CoNLL2003**
SENTENCE	38,219	14,987
TOKENS	912,344	204,567

Table 3: Dataset

2.5. Multilingual Part-of-Speech Tagging with Bidirectional Long Short-Term Memory Models and Auxiliary Loss [5]

Bidirectional long short-term memory(bi-LSTM) networks have recently proven successful for various NLP sequence modeling tasks. The bi-LSTM trained with auxiliary loss. The model predicts the POS and the log frequency of the word. The use of auxiliary loss helps to differentiate the representations of rare and common words. They indeed observe performance gains on rare and out-of-vocabulary words. These performance gains transfer into general improvements for morphologically rich languages. They get accuracy of 96.5 % for english language.

3. Proposed System

The Tagging is a disambiguation task but word are ambiguous and have more then one possible part of speech tags and goal of tagger is to find the correct tag for the given situation or context.

I used two method do the POS Tagging one is Probabilistic and one is Hidden Markov model I will discuss them in sub section in detail.

3.1. Hidden markov model

3.1.1. Introduction. An Hidden Markov Model (HHM) is a probabilistic sequence model, given a sequence of words it computes a probability distribution over possible sequence of labels and choose the best label sequence.

The HMM is based on the Markov chain. The markov chains make the assumption that the if we want to predict the future state in the sequence. we only need the current state.

Given the figure show we have sequence of words and our model gives the sequence of output tags.

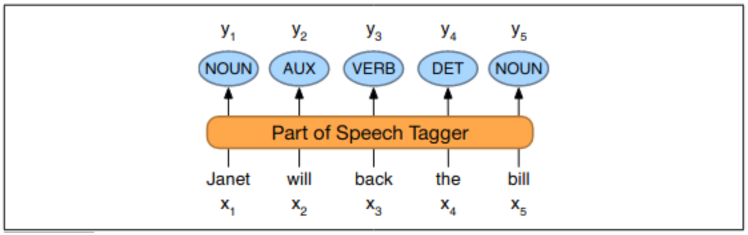

Fig. 2. The Task of POS tagger is mapping input words $w_1, w_2, ...w_n$ to POS tags $t_1, t_2, ...t_n$ [6].

3.1.2. Dataset. We used the Penn tree bank dataset and penn TreeBank part-of-speech tags. following is list of sample Penn tree bank tagset list. full list can be found at their website.

Penn TreeBank tagset		
Tag	**Description**	**Example**
CC	coord. conj.	and, but, or
CD	cardinal number	one, two, 5
DT	determiner	a, an, the
IN	preposition	of, in, by
JJ	adjective	yellow, good
NNP	proper noun	IBM
RB	adverb	quickly
VB	verb base	eat

Table 4: Sample tagset and there Discription

List of all the 46 tags used :

['NNP', ',', 'CD', 'NNS', 'JJ', 'MD', 'VB', 'DT', 'NN', 'IN', '.', 'VBZ', 'VBG', 'CC', 'VBD', 'VBN', '-NONE-', 'RB', 'TO', 'PRP', 'RBR', 'WDT', 'VBP', 'RP', 'PRP$', 'JJS', 'POS', '“', 'EX', '”', 'WP', ':', 'JJR', 'WRB', '$', 'NNPS', 'WP$', '-LRB-', '-RRB-', 'PDT', 'RBS', 'FW', 'UH', 'SYM', 'LS', ''].

3.1.3. Method. An HMM is specified by the following components:

$Q = q_1, q_2, ...q_n$: a set of N states.
$A = a_{11}....a_{jj}...a_{nn}$: a transition probability matrix.
$O = o_1, o_2, ...o_T$: a sequence of T observations.
$B = b_i(o_i)$: a sequcne of observation likelihoods.
$\pi = \pi_1, \pi_2, ..., \pi_N$: initial probabilty distribution over states.

Markove Assumption :$P(q_i|q_1, .., q_i{-}1) = P(q_i|q_i{-}1)$

Transition probability : Transition probability is the probability that is the current tag is t_i given previous tag is t_{i-1} .

$$A = P(t_i|ti-1) = \frac{C(t_{i-1}, t_i)}{C(t_{i-1})}$$

Emission probabilty : probability that the given tag t_i may associated with a given word w_i .

$$B = P(w_i|t_i) = \frac{C(t_i, w_i)}{C(t_i)}$$

The probable tag sequence from the bigram tagger :

$$\hat{t}_{1:n} = \underset{t_1 \dots t_n}{\operatorname{argmax}} P(t_1 \dots t_n | w_1 \dots w_n) \approx \underset{t_1 \dots t_n}{\operatorname{argmax}} \prod_{i=1}^{n} \overbrace{P(w_i|t_i)}^{\text{emission}} \overbrace{P(t_i|t_{i-1})}^{\text{transition}}$$

3.2. Probabilistic model

In Probabilistic model we calcuate the bi-gram probabilty for the tags and probability the word have tag t_i .

Transition probability :

$$A = P(t_i|ti - 1 = \frac{Count(t_{i-1}, t_i)}{Count(tt-1)}$$

Emission probablity :

$$B = P(w_i, t_j) = \frac{Count(w_i, t_j)}{Count(w_i)}$$

The probable tag for word w_i :

$$tag(w_i) = argmax(A \times B)$$

4. Results

Accuracy :

Accuracy = Total correct words tagged / total words tagged

Probabilistic model gives around 87.5 % accuracy on sample test data.

HHM based model gives around 51.3 % accuracy on sample test data.

5. Conclusion & Future Work

I will try to rectify the Hidden markov model because it is not giving accuracy as expected and add the name entity recognition using Conditional random fields for the penn tree bank dataset.

References

[1] T. Brants, “TnT – a statistical part-of-speech tagger,” in *Sixth Applied Natural Language Processing Conference*. Seattle, Washington, USA: Association for Computational Linguistics, Apr. 2000, pp. 224–231. [Online]. Available: https://aclanthology.org/A00-1031

[2] J. Giménez and L. Màrquez, “SVMTool: A general POS tagger generator based on support vector machines,” in *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*. Lisbon, Portugal: European Language Resources Association (ELRA), May 2004. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2004/pdf/597.pdf

[3] M. Collins, “Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms,” in *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*. Association for Computational Linguistics, Jul. 2002, pp. 1–8. [Online]. Available: https://aclanthology.org/W02-1001

[4] X. Ma and E. Hovy, “End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 1064–1074. [Online]. Available: https://aclanthology.org/P16-1101

[5] B. Plank, A. Søgaard, and Y. Goldberg, “Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 412–418. [Online]. Available: https://aclanthology.org/P16-2067

[6] D. Jurafsky and J. H. Martin, *Speech and Language Processing (2nd Edition)*. USA: Prentice-Hall, Inc., 2009.