"Part of Speech Tagging – A Corpus Based Approach"

S. Rashmi(✉) and M. Hanumanthappa

Department of Computer Science and Applications,
Bangalore University, Bangalore 560056, India
{rashmi.karthikl23,hanu6572}@bub.ernet.in

Abstract. POS tagging, an ideal way to augment a corpus is an imperative abstraction for text mining. However with an increase in the amount of linguistic errors and distinctive fashion of language ambiguities, the data filtered by POS tagging is noisier. In this paper, probabilistic tagging and tagging based on Markov models are combined to estimate the association probabilities. Based on this combined approach, error estimation model is defined. Comparison study is made on different corpus available in NLTK such as Crubadan, Brown and INSPEC. The results obtained by the proposed methodologies show a drastic increase in the accuracy rate of about 98% when compared to the existing algorithms which shows an average of 96% accurate. The performance measure is plotted to calculate the error ratio across the maximum-likelihood estimation.

Keywords: Part-of-Speech tagging (POS Tagging) · Hidden Markov Model (HMM) · Error estimation · Natural language processing

1 Introduction

Tagging defines an association. POS tagging tags Part-Of-Speech labels for a token in any language structure. POS is immensely used in text mining and it is an unsupervised/supervised classification. If a tagger is supervised then it counts the number of labeled set of data whose efficiency relies on the existence of the tagging dictionary. On the other hand unsupervised tagging seems to be a reasonable solution because they use un-annotated language models. The efficiency of any POS tagger depends on series of criterion as mentioned below

- Syntactic/Grammatical structure of the sentence should be correct. If there are any syntactical mistakes in a given sentence this could accelerate a chaos in an absolutely efficient POS tagger.
- When supervised model is used, the differences in working fashion, font styles, or the embodied texts between the input data and labeled/training data substantially drops the robustness of the system.
- Ambiguities in certain tenses in English language. For example, should the words such as 'have', 'be', 'book' and 'keep' be treated as verbs or their base/own forms?

A. Unal et al. (Eds.): SmartCom 2016, CCIS 628, pp. 88–96, 2016.
DOI: 10.1007/978-981-10-3433-6_11

- If a probabilistic measure is used for tagging then how much should be the support threshold? How to determine the base dependency of a tagger for a probable architecture?

To domicile the above issues, we propose a probabilistic tagging model which is data-rich and it overcomes the data scarcity encountered in unlabelled data sets. Intuitively the model proposed establishes a tag-engagement for each word of the operational data against the trained dataset. In the next step Markov models are used to visit the node probability of the hidden tags. A rigid performance evaluation is interfered to study the transitions among the labeling and the encoding of the tag-determiners [POS of Standard English].

2 Literature Review

Recent works in the field of NLP have reached a new horizon. There are multiple text processing system that shows an outrageous improvement in terms of accuracy and efficiency. Various researchers have proposed elementary methods to address the problem of POS tagging. Penn Treebank for English corpora was majorly used for the study of POS annotations. Depajan Das et al. [1] work on annotated corpus by building a dictionary. The word alignment exhibited in this work makes use of parallel data for label propagation. The scarcity of standard language corpora and the language challenges of defining a manual-tag construction are studied by Griffins et al. [2].

In the recent study made by Yoong Keok Lee et al. [3] an unsupervised POS tagging is constructed. The regular distributions of language projections is enforced for this study however the method falls costly and not suitable for many systems in a diverse range of applications. In [4], authors have shown a elegant way to enrich the corpus by showing labeling across ach word then assigning the tagged labels for a new word. The tagging schemes shown here make fair grammatical distinctions and hence as a result a large data set is formed. This is found helpful for inherit category of applications. Authors Leon et al. [5] has proposed a tagger using bootstrap constraints for unlabeled data. This is integrated with the probabilities of the existing-label tokens. An accuracy of 88.7% for tagging is achieved. The work also imposes the tagging issues related to a twitter text. Hand-annotated tagging is shown by the authors Clark and Ritter [6].

Reasonably, the classifier model designed with combinational machine learning models seems to provide greater results that transcend the ability of a human to tag the corpus. With this in mind, we have worked on uncertainties viewed in the existing literature. To overcome the problems associated with annotation, we decided to do the automatic tagging using Markov models and in scenarios where the tagging is not possible using the above said model, probabilistic measures are used. Subsequently the performance is enhanced due to the amalgamation of appending new features in tag-rule table. In our work we have shown how POS can be considered as a classification problem. Two lists are created; one says the available tags and the second talks about the POS tags taken as training set. A "HMM-POS [Hidden Markov Model – POS]" is an algorithm proposed in our work. This tags the POS for various lexicon units.

3 Our Approach – An Overview of the Methodologies

POS tagging can be achieved when it is viewed as a classification problem. Training set and positive tags are the two major considerations for this. However the problem of POS tagging is quite not simple and there are three prime elements for this deliberation.

The performance of a POS tagger is influenced by evaluation paradigm and an assessment criterion. The effectiveness of a tagger is relied upon the tagset. Tagset includes the tagging rules which can be projected as a statistical measure. In order to define a tagset either raw or annotated corpora is required. The granularity of the tagset is the main prerequisite as the higher granularity increases the accuracy of the tagger also increases. Perhaps this will chop off some areas and takes to count only the major consideration. In Fig. 1, the POS tagging model is shown. The figure shows the creation of tags and tags rules with the help of the statistical models such as entropy structure, randomization over conditional field, Hidden Markov model. These models are mainly used for POS tagging, Word Sense Disambiguation (WSD) and other areas of research interest under NLP.

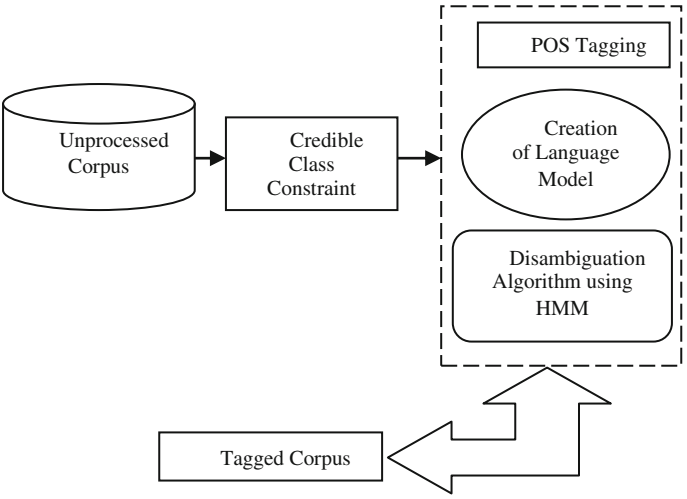

Fig. 1. A POS, our approach

3.1 Hidden Markov Model Based Tagging Approach

Hidden Markov Model (HMM) us a statistical language model that helps to define a time data. It is a process of converting an unobserved/unknown state into an known state. In POS tagging problem, the most appropriate and probable tag/label is selected for an input sentence by observing the previously marked labels in the chosen corpus. The tag sequence is chosen to maximize the probability as shown in equation [1].

$P(W_n|T).P(T|T_n) - -[1]$. The Current word W_n is allotted the tag (T) by observing the tag (T) as with the previous 'n' tags.

Taggers are usually developed using some of the types or approaches described below:

Tag with Stochastic Model: This evaluates the maximum likelihood of the observed tagset. HMM follows the below structure

$$present\_state(DET - N) > present\_state(DET - DET)$$

Tagging using Association Rules: Here the maximum likelihood of the tagset is determined using the predefined the association rules as shown below

$$if < any\_pattern >$$
$$then < Some\_pattern >$$

The prime approaches to be adopted while defining the tagging are, (1) HMM based tagging where all the prevalent information is used and to manoeuvre for deriving the assumption. Calculate the accuracy to study the integrity of this assumption. This approach is called bold approach. (2) Tagging based on constraint grammar. Here tagging assumption/hypothesis is not made as in (1) however the impossible/irrelevant tags are removed. This is called as cautious approach. (3) Tagging deployed on adaptation. A hypothesis is made and later can be changed if need arises. This kind of approach is also called as whimsical approach. In this section an algorithm based on HMM structure used for tagging is described. In this approach we choose the most likely tag for a given sentence where a POS can be derived. The criterion for searching a particular tag is given below.

$$P(n^{th}\_word - bit|tag).P(tag|tags\_of(n-1)\_word\_sequence)$$

The tag sequence that maximizes the above condition is chosen as the most likely tag out of all the possible tags.

3.2 Hidden Markov Model Based Tagging Algorithm

The algorithm (Fig. 2) described in this section explains how tagging is done using HMM structure. The below algorithm is explained by considering two examples (Ex.1 and Ex.2) as stated below

I/PRP have/VBP to/TO book/VB a/DT flight/NN today/NN ——————Ex.1

The/DT book/NN series/NNS of/N Spider/NN man/NN are/VBP really/RB good/JJ————Ex.2

- Consider all the words in the above sentences (Ex.1 and Ex.2) are tagged. Let us consider the word 'book' is not tagged. Perform the step3 from the above algorithm.

to/TO book/? For Ex.1

The/DT book/? For Ex.2

- Possible tags for the word 'book' are NN and VB

step1:Input_the_word_sequence_which_
consists_of_unpragmatic_observations

step2:Assign_the_tags_for_the_known_
word_sequence

step3:for_a_bigram_transition_select_the
_the_tag-(t_a)for_word-(W_a)
Note:Only_when_known_observation_is
_encountered.
Observe_the_tag-t_{a-1}and_compare_with_(W_a)
i.e.,t_a=argmax$_b$P($t_b|t_{a-1}$,W_a)---(1)

step4:Now_in_HMM,make_an_hypothesis
i.e.,t_a=argmax$_b$P($t_b|t_{b-1}$)P($W_a|t_a$)---(2)
where_P($t_b|t_{b-1}$)indicates_present_tag_sequence
P($W_a|t_a$)indicates_the_maximum_likelihood_pair
repeat_step3_to_step4_for_all_word_sequence

step5:Pragmatic/tagged_observations_are_
deduced_for_all_wordpair_in_the_sentence

Fig. 2. HMM based POS tagging algorithm

- Apply (2) of step4 from the algorithm.

  We choose the maximum likelihood of the tag which is seen as described here.
  $P(NN|TO)P(book|NN)\_or\_P(VB|TO)$
  $P(NN|TO)P(book|NN)\_or\_P(book|VB)$

Phase 1: $P(NN|TO)$ and $P(VB|TO) - -(a)$ - In this phase we scan the corpus to find the above probabilities as shown in (a). In reality the probability of 'to' preceding verb is more (example: to race, to fly, to eat…). However these are the sentences which the word 'to' precede a noun as well (example: run to temple, come to house, go to school…). Therefore calculate the likelihood ratio in the corpus. In our experiments we have considered three major available in NLTK. Those are Crubadan, Brown and INSPEC. The probability ratio of (a) is studied in each of these corpus and results obtained are tabulated in the Table 1.

Phase 2: $P(book|NN)$ and $P(book|VB) - -(b)$

Table 1. Occurrences of TO as NN & VB in three corpuses

	$P(NN\|TO)$	$P(VB\|TO)$
BROWN	0.024	0.36
CRUBADAN	0.061	0.13
INSPEC	0.072	0.45

Table 2. Occurrences of BOOK as NN & VB in three corpuses

	$P(book\|NN)$	$P(book\|VB)$
BROWN	0.00043	0.00004
CRUBADAN	0.00037	0.13
INSPEC	0.072	0.45

As observed in Phase 1 the likelihood of 'to' being VB is more and hence Phase 1 talks about the maximum probability of the word 'to'. In this phase we estimate the maximum likelihood probability of book as NN or VB. Tests were conducted on the same three corpus. The results of this test are shown in Table 2. From Table 2 it is observed that the word 'book' has the maximum probability of being NN rather that VB as in all the three corpuses this is evident. Finally we combine the probabilities of Phases 1 and 2 by considering the highest probable factor. This evaluation is shown in the Table 3.

As shown in Table 3 the probability of the word 'book' being VB is more than NN.

Table 3. The probability of the word 'book'

	$P(VB\|TO)P(BOOK\|VB)$	$P(NN\|TO)P(BOOK\|NN)$
BROWN	0.00043	0.00004
CRUBADAN	0.00037	0.13
INSPEC	0.072	0.45

Therefore in the final stage the tag VB is assigned as POS for 'book'. Phases 1 and 2 explains how POS is done for a single word however for the multiple word sequence, the same paradigm is used but has to be elaborated using Bayesian theorem. This is portrayed in Phase 3.

Phase 3: In this phase all the word pair in a given sentence is tagged to their tag sequence based on the likeliness parameter. In order to do so, Bayesian theorem with chain rule is applied. Consider a set word sequence pair W_1, W_2 … W_N and a tag-set sequence T_1, T_2 … T_N. Consider the tag-rule to be of the form γ. According to the Bayes' rule, we have,

$$
\begin{aligned}
&T \in \gamma \\
&\hat{T} = \arg\max\_P(T|W) \\
&\hat{T} = \arg\max_{T\in\gamma} \frac{P(T)P(W|T)}{P(W)} \\
&\hat{T} = \arg\max_{T\in\gamma} P(T)P(W|T)
\end{aligned}
$$

According to the chain rule,

$$P(X,Y) = P(X|Y)P(Y) = P(Y|X)P(X)$$
$$P(X,Y,Z) = P(Y,Z|X)P(X) = P(Z|X,Y)$$
$$P(Y|X)P(X)$$
$$P(W,X,Y,Z) = P(W)P(X|W)P(Y|X,Y)$$
$$P(Z|X,Y,Z\ldots)$$

$$P(T)P(W|T) = \prod_{i=1}^{n} P(W_i|W_1T_1\ldots..W_{i-1}T_{i-1}T_i)P(T_i|W_1T_1\ldots..W_{i-1}T_{i-1}T_i)$$

Present Tag **Tag History** -- (A)

$$P(T)P(W|T) - P(W_i)P(W_2|W_1)P(W_3|W_2W_1)\ldots.$$

Trigram approximation can be deduced by using following condition,

$$P(T)P(W|T) - P(W_i)P(W_2|W_1)P(W_3|W_2W_1)\ldots. \quad \text{(a)}$$
$$P(W_i|W_1T_1\ldots..T_{i-1}T_i) = P(W_i|T_i)$$

With trigram approximation (a), the trigram assumption can be made based on the recent two probable states in addition to the present. This is indicated in the (b)

$$P(T_i|W_1T_1\ldots..T_{i-1}) = P(T_i|T_{i-2}T_{i-1}) \quad \text{(b)}$$

By considering (a) and (b) for (A), we get,

$$P(T)P(W|T) = [P(T_1)P(T_2|T_1)\prod_{i=3}^{n} P(T_i|T_{i-2}T_{i-1})\prod_{i=1}^{n} P(W_i|T_i)]$$

4 Results and Discussions

In order to evaluate our result the interface using Natural Language Tool Kit (NLTK) was developed. The tool kit provides all the linguistic features to implement the required the algorithm. The result was evaluated on the three corpuses namely CRU-BADAN, INSPEC, & BROWN. An accurate means of calculating the efficiency is by Recall and Precision. However each word is associated with utmost one tag for any given instance, calculating the F-measure score will not make good sense. Perhaps one can calculate the recall and precision for individual tag, for e.g. Recall and Precision for NN variations. For every tag encountered in the training data-set – The tagging associated with each word in the chosen corpus, three catalogues are maintained. These are True Positive (TP), False Positive (FP) and False Negative (FN). If the tag of trained dataset and test dataset match then increment the value of TP by 1. If there is no match between them we increment FN for the actual/original tag and also FP for those

tags that our proposed algorithm mistakenly chose. With the obtained values, Recall and Precision can be calculated. The size of the corpus in terms of word ratio is approximately 10,000 words. We have Recall, which defines the likelihood of identifying the positive samples. This can be viewed as the proportion of every positive test samples which is modeled correctly. Recall is computed using the formula (i)

$$Recall = \frac{TP}{TP + FN} \tag{i}$$

The precision talks about the hypothesis and assumptions. This is defined as being proportionate for every positive prediction that made on the test sample. Prediction can be calculated using the formula (ii)

$$Precision = \frac{TP}{TP + Fp} \tag{ii}$$

Finally we arrive at F-measure. This defines the mean of recall and precision approximated by eq. (i) & (ii). F-measure is given by,

$$F - Measure = \frac{1}{\alpha \frac{1}{P} + (1 - \alpha) \frac{1}{R}} \tag{iii}$$

The component α is called balance co-efficient. The value of α is considered as 0.5. This value can be changes conventionally. With the proposed algorithm, we arrive at the following values of Recall, Precision and F-measure that was obtained applying (i), (ii) & (iii) as indicated in the Table 4.

Table 4. Calculation of F-Measure with the values of Recall and Precision

Recall	0.98
Precision	0.97
F-Measure	**0.98**

5 Conclusions

POS tagger, a very important criterion for language analysis is often found in many variations but performance is a major issue with these taggers. With the advent of many POS tagging the accuracy has become a major challenge. Hence in this work, a novel POS tagger is defined adopting the features of HMM. To prove the accuracy of our system, three corpuses have been chosen. We showed how efficient can be increased. The results show that the recommended approach is about 98% efficient. The drawback of this system is the data processing time. This is directly proportional to the size of the input. Since the corpuses chosen for our experiments are huge, time of about 20 s was incurred to arrive at the output. Further extensions to this are comprised of defining the complex tags to increase of the speed of the tagger. In addition to this, one can define a morphological analyzer for multiple natural languages. POS tagging rule can be prepared beforehand. This helps the speed and also the accuracy.

References

1. Das, D.: Unsupervised part-of-speech tagging with bilingual graph-based projections. In: The 49th Annual Meeting of the Association for Computational Linguistics, Portland, Oregon, USA, pp. 600–609, June 2011
2. Goldwater, S.: A fully Bayesian approach to unsupervised part-of-speech tagging. In: Association for Computational Linguistics, vol. 45, p. 744 (2007)
3. Lee, Y.K.: Simple type-level unsupervised POS tagging. In: Association for Computational Linguistics Conference on Empirical Methods in Natural Language Processing, Cambridge, MA, pp. 853–861, October 2010
4. de Gruyter, W.: Corpus Linguistics: An International Handbook, vol. 1, ISBN 978-3-11-021142-9
5. Derczynski, L.: Twitter part-of-speech tagging for all: overcoming sparse and noisy data. In: Recent Advances in Natural Language Processing, Hissar, Bulgaria, pp. 198–206, pp. 7–13, September 2013
6. Ritter, A.: Named entity recognition in tweets: an experimental study. In: Association for Computational Linguistics Conference on Empirical Methods in Natural Language Processing, pp. 1524–1534 (2011)