POS Tagging For Resource Poor Indian Languages Through Feature Projection

Pruthwik Mishra[1] **Vandan Mujadia**[2] **Dipti Misra Sharma**[3]
Language Technologies Research Center, IIIT Hyderabad
Kohli Center On Intelligent Systems
pruthwik.mishra@research.iiit.ac.in,vmujadia@gmail.com,dipti@iiit.ac.in

Abstract

We present an approach for POS tagging with out any labeled data. Our method requires translated sentences from a pair of languages. We used feature transfer from a resource rich language to resource poor languages. Across 8 different Indian Languages, we achieved encouraging accuracies without any knowledge of the target language and any human annotation. This will help us in creating annotated corpora for resource poor Indian languages.

Keywords

POS, NLP, corpus, parallel corpora, Feature Transfer, Alignment, Mapping

1 Introduction

Part-Of-Speech(POS) (Bharati et al., 2007) Tagging is considered as a preliminary task for various Natural Language Processing(NLP) tasks. POS Tagging primarily assigns class labels to words based on some extracted features. The POS tagged corpus can further be used for parsing, building lexical dictionaries, frequency lists and many more subsequent tasks [1]. For automatic POS tagging, the state-of-the-art POS taggers use large POS annotated data sets and try to learn the appropriate class labels for words depending on various hand annotated features. There are many Indian Languages which are unexplored due to the unavailability of annotated corpora. But recently, there has been a lot of efforts to create monolingual as well as bilingual corpora for different Indian Languages.

Hindi is resource rich in this regard as there are many linguistic resources created for Hindi. One of the notable corpus available is the Hindi Treebank (Bharati et al., 2006). Statistical POS taggers trained on Hindi Treebank data for Hindi achieved around 93% accuracy (Gadde and Yeleti, 2008). Stochastic or Statistical Taggers are also used for Indian languages like Punjabi, Urdu, Marathi, Telugu, but their accuracies fall due to lack of large annotated corpora. POS Taggers for other Indian Languages have not been evaluated. So these languages are resource poor in terms of high quality linguistic annotated data.

The motivation behind this work is to create lexical resources for resource poor Indian languages. All the Indian Languages are morphologically rich, prefixes and the word ending suffixes encode a lot information about the category of the word. We try to leverage these similarities and availability of Hindi corpus for creating resources for other languages.

The paper is divided as per the following. In the section 2, we describe Background of Projection using aligned Corpora. Section 3 gives an account of Corpus Details, section 4 describes the Algorithm and various tools used. Section 5 presents the Experimental Results and in the subsequent section 6, we have the error analysis. The future work is discussed in the concluding section.

2 Background

Many supervised learning techniques reported state-of-the-art accuracy of around 90% for POS tagging in Indian Languages. POS Tagging is more accurate in most of the Indo-Aryan Languages while the results are poorer

[1] http://www.ahds.ac.uk/creating/guides/linguistic-corpora/

Proc. of ICON-2017, Kolkata, India. December 2017 ©2017 NLPAI, pages 50–55

for agglutinative Dravidian Languages. But one major bottleneck in POS tagging is the requirement of a large labeled corpus which is difficult to create. To overcome this difficulty, many researchers have employed unsupervised techniques which are less accurate (Accuracies reported in the range of 70-80%) (Christodoulopoulos et al., 2010)

So in our approach, we leveraged the gold quality corpus of a resource rich language and transferred features to a resource poor language. The only resource available to us is a parallel corpus with the resource rich one. We report the results using two different tag-sets - one being the tag-set defined for Indian Languages named as the IIIT - tagset (Bharati et al., 2006) which is fine-grained and the Universal Tag-set (Petrov et al., 2011) mostly used for Unsupervised and semi-supervised POS-Tagging which is coarse-grained. We evaluated on 8 Indian Languages and obtained overall average accuracies of 81%.

(Yarowsky et al., 2001) introduced robust projections across aligned corpora. They used a statistical POS tagger for tagging source side text and transferred the POS tags to the target side from the word alignments obtained. The noisy transfers were filtered out re-estimating the the most frequent tag sequence model. Other works (Das and Petrov, 2011) used bilingual projections using Universal Tag-set. All these methods employ Label Propagation (LP) to transfer the tags from labeled data to unlabeled data. These are examples of semi-supervised techniques and major work has been done on European languages. The work of (Das and Petrov, 2011) is closest to (Yarowsky et al., 2001). These methods are evaluated on data sets which are very similar to the parallel corpus. Direct transfer of tags using raw projection can lead to very noisy POS tags. Instead of directly matching words from the word alignments available; we use the feature of the words which are clear indicators of POS tags realized through rich morphology. Because of the limited size of bitext (parallel corpus) chances of finding exact matching words gets reduced. We do not use the observed word as a feature. For avoiding non-matched features, we use back-off smoothing. Thus we have an approximate feature representation for any word occurring in the corpus. The features used suffice to the back-off model. The Suffixes and prefixes provide valuable cue in the identification of a particular POS category. Additionally, suffixes help to disambiguate between various similar categories.

Language	Domain	#Tokens
Hindi	Health	368K
Hindi	Tourism	474K
Marathi	Health	382K
Marathi	Tourism	278K
Konkani	Health	346K
Konkani	Tourism	328K
Urdu	Health	371K
Urdu	Tourism	473K
Bengali	Health	300K
Bengali	Tourism	387K
Gujarati	Health	329K
Gujarati	Tourism	388K
Punjabi	Health	386K
Punjabi	Tourism	425K
Tamil	Health	313K
Tamil	Tourism	312K
Telugu	Health	316K
Telugu	Tourism	316K
Malayalam	Health	286K
Malayalam	Tourism	291K

Table 1: ILCI Corpus Details

We trained the models on general domain and tested on health domain data. The performance of our models is comparable to the state-of-the-art systems in out-of-domain data.

3 Corpus Details

We used two data sets for our experiments.

1. ILCI (Indian Languages Corpora Initiative) parallel corpora
2. Hindi Tree-bank

The data used for parallel corpora was the ILCI (Choudhary and Jha, 2014) corpora released for different languages. We have experimented on 8 languages :- Punjabi, Konkani, Bengali, Telugu, Malayalam, Urdu, Marathi, Gujarati

The details of the ICLI corpus are shown in Table 1, the number of sentences in each

Data-Set	#Sentences	#Tokens
Hindi Treebank	21K	450K

Table 2: Hindi Treebank Details

language was 25K.

The Hindi Treebank (Bharati et al., 2006) creation task was taken up at IIIT, Hyderabad. The treebank is a multi-layered representation of sentences with syntactic and semantic annotation. The syntactic annotation includes morph analysis, POS Tagging, Chunking of words or tokens occurring in a sentence. We used Hindi treebank for ensuring high quality projection of POS tags. The Hindi Treebank is annotated with POS tags from IIIT tag-set(tagging annotation guidelines described in (Bharati et al., 2006)). The details of the Hindi Treebank is presented in Table 2. We also converted IIIT tags to Universal tags (Petrov et al., 2011) and evaluated the POS Tagging accuracy for both the tag-sets. Universal Tag-set is often used for projection techniques to remove ambiguities related to finer grained tags.

4 Algorithm

Our approach is an example of feature representation transfer. We transfered the knowledge acquired from a language to another language. In this paper, the source language used for feature transfer was Hindi and the target languages for projection were resource poor Indian Languages explained in the above section. The scarcity of data for any language will not impact the performance if a huge training data set is available for another language. With a mapping between the feature sets of the concerned languages, we have the luxury of creating training data of comparable size for a resource poor language.

Our algorithm has 5 steps.

4.1 Word Alignment

Learning word alignments from the parallel text is the first step in our approach. We used GIZA++ tool [2] for capturing the word level alignment between sentences that are aligned for a pair of languages. The raw text files for a source language and target language serve as the inputs for the tool. In this case, the source language was Hindi and target language was any of the resource poor Indian languages. GIZA++ tool finds the alignments between words with translation probabilities. It also generates files with translation probabilities of aligned sentences. A word can have multiple alignments, but we selected the alignment with highest probability. We were able to eliminate noisy alignments by only selecting the most likely alignment for a word.

Feature	Example
Prefix length 1	प
Prefix length 2	पत
Prefix length 3	पत्
Prefix length 4	पत्र
Prefix length 5	पत्रक
Prefix length 6	पत्रका
Prefix length 7	पत्रकार
Suffix length 1	◌ें
Suffix length 2	◌ों
Suffix length 3	रों
Suffix length 4	◌ारों

Table 3: Features for Hindi

4.2 Feature Selection

As POS Tagging is sequence labeling task, certain features need to be captured in classifying the words and assigning them appropriate tags. Indian Languages are morphologically rich, therefore prefixes and suffixes provide a lot of information about the category of the word. For Indian Languages, we considered the following morph features:-

- The prefix characters up to 7 characters
- The suffix characters up to 4 characters
- Length of the word
- Context Window size of 3 (Previous word, Current word and Next word)

For example the feature representation for a Hindi word पत्रकारों (Patrakāron - Journalists) is given in Table 1:

The above features are extracted from the words present in the Hindi Treebank. After this step, all the words in the Hindi Treebank are represented in terms of their features.

[2] https://github.com/moses-smt/giza-pp.git

```
if LENGTH($word$) < LENGTH($feature$) then
▷ The prefix length can vary from 1-7 and suffix length can range from 1-4
    $feature \leftarrow NULL$
```

Feature	Source	Mapped
Prefix 1	व	ਵ
Prefix 2	वि	ਵਿ
Prefix 3	विव	ਵਿਆ
Prefix 4	विवा	ਵਿਆਹ
Prefix 5	विवाह	ਵਿਆਹੀ
Prefix 6	विवाहि	ਵਿਆਹੀਆ
Prefix 7	विवाहित	ਵਿਆਹੀਆਂ
Suffix 1	त	◌ਂ
Suffix 2	◌ित	ਆਂ
Suffix 3	हित	◌ੀਆਂ
SUffix 4	◌ाहित	ਹੀਆਂ

Table 4: Feature Mapping Between Hindi and Punjabi

4.3 Mapping File Creation

After obtaining word level alignments, we created feature level mapping files based on the features defined in the previous subsection. As the word alignments with highest probabilities were taken into account, the corresponding feature files supported the best possible mapping from the source language to the target language. The example of Hindi - Punjabi pair mapping file is given in Table 4. The word in Hindi is 'विवाहित' (Vivāhita - married) and the aligned word in Punjabi is 'ਵਿਆਹੀਆਂ' (Vivāhita - married). We did not normalization the text before the feature transfer. This was done keeping in our effort of not including any language specific information for any resource poor language. If the same feature got mapped to multiple target features, we selected the target feature with highest probability. There are 7 features corresponding to prefixes and 4 features for suffix, so there are total 11 feature mapping files from Hindi to one of the resource poor languages.

4.4 Feature Transfer

This is the most vital step of the algorithm in which the features obtained from the tokens present in the Hindi Treebank are transferred to other languages. Hindi Treebank data is used to avoid noisy projections from Hindi to resource poor languages. Each word in the treebank is represented by the features described above. From the mapping files, we obtain a particular Hindi feature and its corresponding mapped feature in the language under test. If a feature is missing from the feature mapping file, a back-off model finds the next lower length feature. These feature transformations are essential for a sound representation in the target side. Otherwise, in case of features that are not-found, the target side would have been filled with NULL features and will affect the performance of the POS Tagger.

4.5 Creation of Training Model

This step is the training phase of POS Tagging where the samples are assigned pre-defined labels i.e. POS tags. We have used Conditional Random Fields(CRF) classification algorithm for our task. The conditional random fields are implemented via CRF++ [3]tool. This tool receives features in the form of a template. From the training samples, CRF creates a model assigning feature weights to the individual features. After the model is created, a test data set is used to predict the POS tags of the test samples.

The whole process can be modeled as a composition of different functions. The final model for predicting POS tags is given in equation 1

$$final\_model = M(T(f(A(x, y)))) \qquad (1)$$

where A→ Alignment between word x and y
f→ Feature Representation for the words x, y
T→ Transfer of Features from x to y
M→ Model creation using the transferred features

5 Experiments & Results

The experimental results are shown in Table 5. The 1st entry for Gujarati corresponds to the word alignments obtained from Original Hindi - Gujarati ILCI parallel corpus. The other entry reflects the accuracy obtained after preprocessing the parallel corpus. Case markers are not present as separate tokens in Gujarati. As a result of this, there were no alignments for many post-positions used as case markers

[3] https://taku910.github.io/crfpp/

Language	Accuracy	#Train	#Test
Gujarati	80.5	450K	6.5K
Gujarati	86.2	380K	6.5K
Punjabi	84.3	450K	9K
Urdu	85.6	450K	7.5K
Konkani	76.9	380K	7.5K
Bengali	75.5	380K	7.5K
Telugu	74.2	380K	7K
Marathi	77.7	380K	7K
Malayalam	65.01	380K	6K

Table 5: Accuracies for Different Languages

in Hindi. So we combined the post-positions with the preceding noun on the Hindi side and found word alignments from the changed parallel corpus.

6 Error Analysis

Languages which are close to Hindi gave better results than other languages. Gujarati, Urdu and Punjabi have similar syntactic structure to Hindi with minor variations. Gujarati does not separate case markers as Hindi, where the case markers are present as a suffix in the nouns. Other languages like Bengali, Konkani, Marathi, Telugu, Malayalam are morphologically richer than Hindi. So the word level alignment is less accurate. To overcome this shortcomings, we combined the post positions marking the case, the auxiliary verbs with the preceding head categories. The head category in the former was the noun and the corresponding head in the latter was the verb. By incorporating these changes, we were able to capture the inherent syntactic behavior of these languages. The increase in accuracy for Gujarati showed this. This heuristic helped to create a better feature mapping (Petrov et al., 2011).

The sources of major errors are listed as follows:

- Errors in the Gold Data
- Ambiguities between Nouns and Adjective
- Ambiguities between particle and conjunction
- Ambiguities between Verbs and Auxiliary Verbs
- Alignment Issues

7 Future Work

We would extend this work to other resource poor Indian languages. Semantic Clustering using neural networks is an interesting area to explore. As the efficiency of the system relies heavily on the word alignments between a pair of languages, new methods can be implemented to improve the alignments. We will experiment with other sequence labelers like structured perceptron (Collins, 2002), SVM-Struct (Tsochantaridis et al., 2004), sequence-to-sequence learners(Sutskever et al., 2014). We will explore (Das and Petrov, 2011) the label propagation of tags between languages.

Acknowledgement

We thank DIT for providing ILCI corpus which enabled us to carry out the research. We would also like to thank people of different languages for helping out in the error analysis of POS tagging.

References

Akshar Bharati, Rajeev Sangal, Dipti Misra Sharma, and Lakshmi Bai. 2006. Anncorra: Annotating corpora guidelines for pos and chunk annotation for indian languages. *LTRC-TR31*.

Akshar Bharati, Rajeev Sangal, and Dipti M Sharma. 2007. Ssf: Shakti standard format guide. *Language Technologies Research Centre, International Institute of Information Technology, Hyderabad, India*, pages 1–25.

Narayan Choudhary and Girish Nath Jha. 2014. Creating multilingual parallel corpora in indian languages. In *Human Language Technology Challenges for Computer Science and Linguistics*, pages 527–537. Springer.

Christos Christodoulopoulos, Sharon Goldwater, and Mark Steedman. 2010. Two decades of unsupervised pos induction: How far have we come? In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 575–584. Association for Computational Linguistics.

Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 1–8. Association for Computational Linguistics.

Dipanjan Das and Slav Petrov. 2011. Unsupervised part-of-speech tagging with bilingual graph-based projections. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 600–609. Association for Computational Linguistics.

Phani Gadde and Meher Vijay Yeleti. 2008. Improving statistical pos tagging using linguistic feature for hindi and telugu. *Proc. of ICON*.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2011. A universal part-of-speech tagset. *arXiv preprint arXiv:1104.2086*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemin Altun. 2004. Support vector machine learning for interdependent and structured output spaces.

David Yarowsky, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the first international conference on Human language technology research*, pages 1–8. Association for Computational Linguistics.