An Attempt at Multilingual POS Tagging for Tamil

Madhu Ramanathan, Vijay Chidambaram, Ashish Patro
Department of Computer Sciences
University of Wisconsin Madison

Abstract

Part of Speech (POS) tagging is the process of providing every word in a corpus with a syntactic category. In our project we aim to do supervised and unsupervised methods of POS tagging using a multilingual parallel corpora for Tamil, an agglutinative language of ancient Dravidian origin. The multilingual parallel corpora consists of three other languages namely Hindi, Latin, English and French. We experimented on monolingual, bilingual and multilingual corpora using various models and techniques such as the HMM model, SVM model, CRF model and Projection and Probability Re-estimation technique (Yarowsky, 2001) and did a detailed performance comparison in an attempt to capture the properties of the language that aid in increased accuracy for POS tagging. Supervised CRF modeling using a variety of features on a monolingual Tamil corpus revealed that word specific features such as prefixes and suffixes produce an increase of 10% the highest among all combinations of features. Bilingual and multilingual learning shows that the addition of other languages generally produce a decrease in accuracy mainly because of the one to many association among the words while the other reasons being the drop in accuracy produced at every stage of the various pre-processing steps involved in accomplishing the word level pairing. The results of our experiments clearly reflect the relatively free word order and agglutinative nature of the Tamil language and motivates the need for a morpheme based POS tagger to attain a greater accuracy.

1 Introduction

Part of speech (POS) tagging is the process of labeling a part of speech or other lexical class marker to each and every word in a sentence. POS tagging is an essential part of many applications like speech recognition, natural language parsing, information retrieval and machine translation.

Our aim is to perform POS tagging for Tamil which is a Dravidian Language spoken in the Southern part of India which has existed for over two thousand years. Tamil and Sanskrit are considered the two longest surviving classical languages in India, from which the others Dravidian and Indo-Aryan languages have been derived. Tamil also has a rich set of literary works like the Thirukurral which have been manually translated into a number languages. Our aim is to use such parallel corpus and build a method to improve the accuracy of existing taggers that can be used for other applications like automatic machine translation, speech recognition and parsing.

Tamil uses a relatively free word order agglutinative grammar, where suffixes are used to mark noun class, number, and case, verb tense and other grammatical categories.Tamil words consist of a lexical root to which one or more affixes are attached. Most Tamil affixes are suffixes. Tamil suffixes are of two types : derivational suffixes, which either change the part of speech of the word or its meaning, or inflectional suffixes, which mark categories such as person, number, mood, tense, etc.

There is no absolute limit on the length and extent of agglutination, which can lead to long words with a large number of suffixes (Tamil, Wikipedia). Much of Tamil grammar is extensively described in the oldest known grammar book for Tamil, the Tolkppiyam.

The agglutinative nature of Tamil makes tagging a complex process. Various methodologies, both statistical and rule based, have been developed and widely used for POS Tagging in different languages. Tamil being a free form language with a large variety of morphological combinations, inflections and exceptions, developing a rule based method for it would require a lot of effort and also extensive knowledge about the complex grammatical structures which makes it almost impractical. Supervised statistical methods require a large amount of reliable annotated corpus that can be used for training purposes. At the same time a considerable large amount of sentence aligned parallel data (UDHR corpora, Bible corpora, Thirukural corpora, TV news, newspaper articles,etc) are available in a number of languages that we can put to use for this purpose. A large number of those languages such as the European languages have pre-trained POS taggers that can be used to label the text in those languages. Consider these factors we tried to address three main questions:

- When trained on a monolingual corpus what properties/features of the language contribute to increasing the POS tagging accuracy?
- Does the addition of one or more languages from a parallel corpus help in increasing the POS tagging accuracy? If the addition of languages does improve the tagging accuracy then are they any specific properties of the language being paired that lead to an increase in accuracy?

As a means to find the answers to these questions we experimented with monolingual, bilingual and multilingual corpus using various methods such as SVM model, HMM model, CRF model and Bilingual projection and probability re-estimation method (Yarowsky, 2001). The languages that we chose are Hindi, English and French. Tamil follows a SOV word order and we chose Hindi as it a well studied Indian Language with same word order. We also choose two other languages that have the SVO word order namely English and French to see how much the word order property influences the accuracy of the results.

The remainder of the paper is organized into 5 sections. Section 2 deals with the related work, section 3 talks about the method, section 4 about the experiments and analysis and section 5 gives the concluding remarks.

2 Related Work

Tamil is one of the classical Indian languages which has a very strong linguistic base with well defined set of morpho-syntactic rules. However parsing, development of parsing models, chunking, generation of Treebank, POS tagging, morphological analysis, and development of semi-automated and automated tools for these processes in Tamil are at the nascent stage. The existing works on POS tagging is based on morphological analyzers which was built by Vasu Ranganathan (Renganathan, 2001) and Ganesan and RCILTS-T. Due to the constraints, limited coverage of morpho-syntactic and semantic rules, non-availability of methodologies towards large scale development of parsing models, non-availability of standards, non applicability of statistical methods and resource deficiency, reported tools cannot be used directly for all types of NLP applications. These existing tools have been developed using rule based approaches. However, rule based techniques cannot address all inflectional and derivational word forms and peculiar characteristics like relative free word order, syntax with semantics and long distance relationship to a greater extent. Moderate accuracy can only be achieved in rule based techniques. This motivates the need for a statistical approach to POS tagging in Tamil.

Various methods for bilingual POS tagging such as projection and induction have been used to train highly accurate part-of-speech taggers (Yarowsky, 2001) for languages such as Viet-

namese (Dieng, 2003). As one of our methods we use Yurowskys robust projection and probability re-estimation technique to learn the POS tags for Tamil in an semi-supervised manner. There has been some recent work on bilingual (Snyder, 2008) and multilingual learning (Snyder, 2009) where the results show that adding languages generally increases the accuracy when unsupervised learning is done. There has been one attempt at bilingual rule based POS tagger for Tamil using projection and induction techniques that quotes an increase in performance (Selvam, 2009). However, we aim to do a purely statistical approach to POS which does not require any prior knowledge of the grammar rules.

Tag	Description
NN	Noun
CNN	Compund Noun
PRN	Pronoun
CPRN	Compound Pronoun
VRB	Verb
ADJ	Adjective
ADV	Adverb
CONJ	Conjunction
PP	Preposition
NUM	Number
X	Others
P	Punctuation marks

Table 1: Tagset used for Tamil corpus

3 Methodology

We used the Universal Human Rights Declaration corpus (UDHR) which has been translated into over 300 languages for our experimentation (UDHR, UDHR corpus). The UDHR corpus consists of 75 lines of short text translated in all the 300 languages of which we choose the text for our set of languages - Tamil, Hindi, English and French. The following sections describe in detail about the preprocessing step and the monolingual, bilingual and multilingual learning approaches that we experimented with.

3.1 Preprocessing

Before working on this data, we applied a preprocessing step on the data to make it usable for our experiments. We arranged the text by pairing the Tamil text with the other 3 languages. So, we had a total of 3 pair of languages. Sentence alignment was done using Microsoft Researchs Bilingual Sentence Aligner tool (Microsoft, 2003). The sentence aligned files were given to the GIZA++ word aligner and the union method was used to obtain the word alignments (Giza, 1999). The union method was chosen over the intersection that would give a 1-1 pairing because Tamil being an agglutinative language when paired with other languages which do not possess that property would yield very low recall when the intersection method of word alignment was used. The UDHR corpus was a plain text without any POS tagging done for the words. For well studied languages like Hindi, English and French we used existing pre-trained taggers. For Hindi we used the tagger developed by the Society for Natural Language Technology Research and for English and French we used the TreeTagger tool (TreeTagger, 1994). For Tamil, as no such pre-trained tagger was in a usable form we had to hand tag the corpus. Table 1 shows the set of 12 tags used for tagging the Tamil corpus. These tags were chosen as they were the frequently occurring tags that also appear in other languages. We tried to perform this tagging to the best of ability though some errors may have been performed in this step. These tags were used as the gold standard for all our experiments.

3.2 Monolingual Supervised learning

In this method we use the monolingual Tamil corpus alone to perform supervised learning techniques using various methods to estimate the maximum accuracy that can be obtained using a single language and also to find out which features of the language aid in increasing the tagging accuracy. For this purpose we split the dataset into training and test sets. The training set comprised of 80% of the lines while the testing set comprised of 20% of the lines. Since the corpus was small we used 10 -fold cross validation to estimate the accuracies. We trained it using three well known models namely the Hidden Markov Model (HMM), Support vector machines (SVM) and Conditional Random Fields(CRF) .

Strategy	Description
0: one-pass	default strategy
1: two-pass	revisiting results and relabeling
2: one-pass	robust against unknown words
4: one-pass	very robust against unknown words
5: one-pass	sentence-level likelihood
6: one-pass	robust sentence-level likelihood

Table 2: Strategies used in the SVM Model

3.2.1 Hidden Markov Model(HMM)

We used a bigram HMM model along with the viterbi algorithm to train the corpus. Maximum likelihood estimator was used to determine the emission and transition parameters.The transition and emission parameters were calculated as follows:

$$P(t|t') = count(t', t)/count(t')$$
$$P(w|t) = \frac{(count(t, w) + \delta)}{(count(t) + |V| * \delta)} \quad (1)$$

After determining the emission and transition probabilities the probability of a given tag sequence for a given word sequence was determined using the following formula:

$$P(s, w) = \Pi_i(P(t_i|t_{i-1}) * P(w|t_i))$$

3.2.2 Support Vector Machines

We used the SVMtool which is a general POS tagger based on Support Vector Machines to train and test on our corpus. There were several modes of doing the tagging in that tool. Each mode brought a little more complexity into the tagging. We used a set of six strategies to determine the one that gives the maximum accuracy. The six strategies are listed in the Table 2.

3.2.3 Conditional Random Fields

For the conditional random fields we used the CRF++ tool which is a simple, customizable, and open source implementation of Conditional Random Fields (CRFs) for segmenting/labeling sequential data. CRF++ tool allows us to redefine our own set of features. It requires the training and testing files to be in a specific format. It also requires us to define a template file specifying the unigram and bigram features. For every unigram and bigram feature specified in the feature file the tool converts it into a set of binary feature functions associating the specified feature with the output category. Using this tool we built our training and testing files in the required formats and modelled and tested on a variety of combinations of features. The combination of features are listed in Table 3.

Feature	Description
1	Actual word
2	1 Previous Word + Actual word
3	2 Previous words + Actual word
4	2 Previous words + Actual word
5	4 Previous words + Actual word
6	1 Next word + Actual word
7	2 Next words + Actual word
8	3 Next words + Actual word
9	1 Previous word + 1 Next word + Actual word
10	1 Prefix + Actual Word
11	2 Prefixes + Actual word
12	Prefixes + 2 Suffixes + Actual word
13	Prefixes + 4 Suffixes + Actual word
14	Prefixes + 5 Suffixes + Actual word

Table 3: Feature sets used in monolingual learning

From the results obtained, we try to determine the features that give a maximum increase in accuracy for POS tagging.

3.3 Bilingual Learning

3.3.1 Supervised

For the supervised method of bilingual learning we used the same CRF++ tool described above. Tamil was paired with each of the other three languages separately and the tags from the foreign language were projected onto the Tamil words using the word alignments. Then the training and testing files for the CRF++ tool were prepared and the template files were created considering the various combinations of possible features that could affect the accuracy of tagging. The feature sets that we tested on are given in the Table 4.

3.3.2 Semi-Supervised

For this we used the projection and aggressive tag probability re-estimation technique (Yarowsky, 2001). We used POS tag projection from an input language (e.g. English) to Tamil using the word alignments computed during the pre-processing

Feature	Description
1	Actual word + Tag
2	Actual word + Tag + 1 Prev. Tag
3	Actual word + Tag + 1 Next Tag
4	Actual word + Tag + 1 Prev. Tag + 1 Next tag
5	Actual word + (Tag,1 Prev. Tag) pair
6	Actual word + (Tag,1 Next Tag) pair
7	Actual word + (Tag,1 Prev. Tag) pair + (Tag,1 Next Tag) pair
8	Actual word + (Tag,2 Prev. Tags) pair
9	Actual word + (Tag,2 Next Tag) pairs
10	Actual word + (Tag,2 Prev. Tag) pair + (Tag,2 Next Tag) pair
11	Actual word + Tag + 1 Next Tag + 2 prefixes + 5 suffixes

Table 4: Feature sets used in bilingual tagging

	Description
1	Actual word + Tags
2	Actual word + Tags + Next Tags
3	Actual word + Tags + Next Tags + 2 prefixes + 5 suffixes

Table 5: Feature set used for multilingual learning

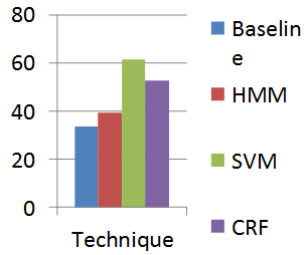

Figure 1: Comparing the highest accuracies obtained by each method with the baseline

stage. Before performing this step, we had to perform some additional pre-processing specific to this technique. For English and French POS tagged input files, we used the POS tag documentation available from TreeTagger to convert the tags to use the same set of coarse-grained tags applied to the Tamil files. Thus, we had a consistent set of tags across English, French and Tamil. This step was performed to enable us to calculate the accuracy of the POS tag projection technique.

As the first step in this technique, we projected the POS tags onto the Tamil file using the word alignments. This gave us an initial set of noisy POS tags for each Tamil word. We then performed an iterative process of aggressively re-estimating the POS tag for each word from the initial set of noisy tags. This was done by truncating the set of least probable POS tags for each word and re-estimating the tag for each word. This technique could not be used for the Tamil-Hindi pairs because there is no documentation available for the Hindi POS tagset used the Hindi POS tagger tool. Thus, we could not map the original set of Hindi POS tags to the our coarse grained set of Tamil tags that was necessary for the sake of calculating the accuracy. So, this technique was used only for the Tamil-English and Tamil-French language pairs. The accuracy was calculated by comparing the final tags estimated by this technique for the Tamil words against the input tags provided by us.

3.4 Multilingual Supervised Learning

For multilingual learning we projected the tags of all the other languages on to the Tamil words based on the word alignment files. The CRF++ tool described above was used again for multilingual learning. The features that gave maximum accuracy in the monolingual and bilingual methods were taken and combined with the feature sets for the multilingual case and the following features set were arrived. Table 5 shows the feature sets used for multilingual learning.

4 Experimentation and Analysis

As our baseline, we considered the most frequent tag in the Tamil corpus (CNN) which gave an accuracy of 33.47%. All our comparisons are done having this as our baseline tagger.

4.1 Monolingual learning

When our corpus was trained on 80% of data and tested on the remaining 20%, the highest of accuracies obtained as a result of our HMM, SVM and

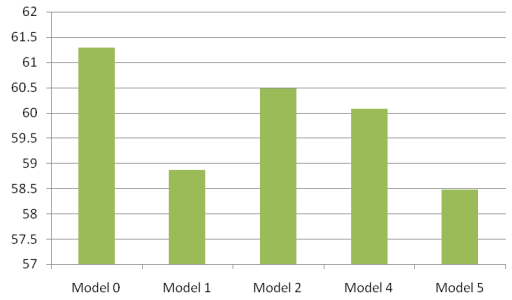

Figure 2: Results obtained by applying various strategies of the SVM model listed in Table 2

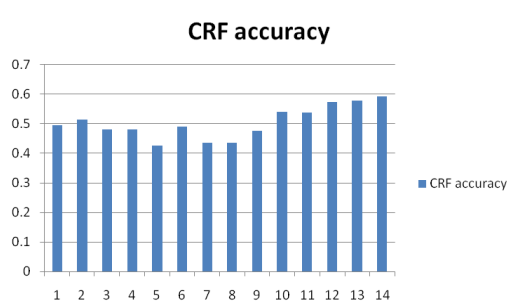

Figure 3: POS tagging accuracy using CRF model for the feature sets listed in Table 3

CRF models are shown in figure 1. It can be seen that SVM model gives the maximum accuracy of 61.29%. It also shows us that HMM model which is a widely used model for POS tagging fails for Tamil given the relatively free word order property it possesses.

Figure 2 shows the accuracies obtained for each of the SVM strategies used and Figure 3 shows the accuracies obtained through the CRF++ tagger for the feature sets described above in table 1. The maximum accuracy of 59.25% is obtained only when the word level features such as prefixes and suffixes are added to the model. This clearly shows that the POS tags are mostly dependent on the word itself than on the previous and next tags or words.

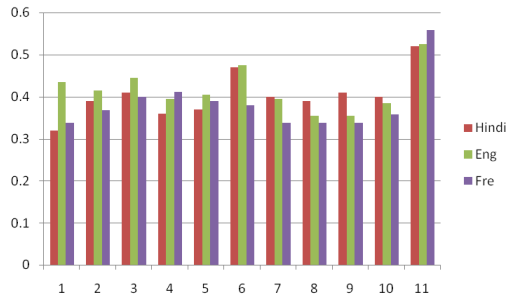

Figure 4: Results of POS tagging using CRF method for the feature sets described in table 4 for a bilingual corpus

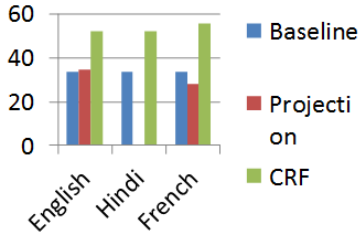

Figure 5: Results of POS tagging using Projection and Aggressive probability re-estimation for a bilingual corpus

4.2 Bilingual Learning

The results of supervised learning using pairs of languages for the feature sets described in Table 4 are given in Figure 4. This shows us that the maximum accuracy is obtained when the model is trained with the current tag and the next tag as features. It also shows us that accuracy on an average is greater for English than for French and Hindi.

Figure 5 shows the results of semi-supervised learning through the projection and aggressive probability re-estimation method for each pair of languages. The accuracy estimates for Hindi are missing because of the lack of proper documentation of the tagsets for the Hindi tagged that we used which was essential to convert the tags into a common tagset. For English the semi-supervised method yields better results than the baseline however the accuracy drops when it is paired with French.

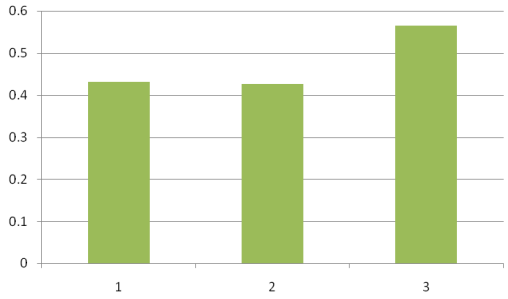

Figure 6: Results of POS tagging for multilingual corpus for the feature sets in Table 5

4.3 Multilingual learning

The results of multilingual learning using the CRF++ model for the features sets listed in Table 5 are given in Figure 6. As predicted the accuracy increases when the features that yield maximum accuracy in the monolingual and bilingual methods are added to the set of all POS tags from the other languages. However, the accuracy is still below the accuracy obtained as a result of the SVM modeling on a monolingual corpus.

5 Conclusion

Now, let us address the questions that we aimed to find answers to at the beginning. Answering our first question we can say that the relatively free word order property does not favor the use of the HMM model for POS tagging. Also, we can see that when language specific features are considered prefixes and suffixes are the ones that give the maximum increase in accuracy. Coming to our second question, the addition of languages does not always produce an increase in accuracy of POS tagging when compared to the monolingual supervised learning methods especially for languages like Tamil with an agglutinative nature. The use of parallel corpus also leads to a drop in accuracy in many of the preprocessing stages such as sentence alignment and word alignment. A possible solution to this would be to split the words into morphemes and then apply the preprocessing steps which we aim to try in the future.

References

N. T. E. J. . B. R. Snyder, B. Unsupervised multilingual learning for pos tagging. In Proceedings of EMNLP (2008).

N. T. E. J. . B. R. Snyder, B. Adding more languages improves unsupervised multilingual part-of-speech tagging: A bayesian non-parametric approach. In Proceedings of NAACL/HLT (2009).

D. Yarowsky and G. Ngai. Inducing multilingual pos taggers and np bracketers via robust projection across aligned corpora. Second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies, 2001, p.1-8.

A. N. M. Selvam. Improvement of Rule Based Morphological Analysis and POS Tagging in Tamil Language via Projection and Induction Techniques.

V. Renganathan. Development of Part-of-Speech Tagger for Tamil, Tamil Internet 2001 conference, 2001.

Language independent part-of-speech tagger. http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/.

Tamil. http://en.wikipedia.org/wiki/Tamil_language.

H. K. Dinh Dieng. POS tagger for English - Vietnamese Bilingual corpus.

UDHR corpus. http://research.ics.tkk.fi/cog/data/udhr/.

Microsoft research bilingual sentence aligner. http://research.microsoft.com/en-us/downloads/aafd5dcf-4dcc-49b2-8a22-f7055113e656/.

Giza ++. http://www.statmt.org/moses/?n=FactoredTraining.AlignWords.