A POS Tagger for Malayalam using Conditional Random Fields

Ajees A P  
Department of Computer Science  
Cochin University of Science and Technology  
Kochi, India

Sumam Mary Idicula  
Department of Computer Science  
Cochin University of Science and Technology  
Kochi, India

Abstract—POS tagging is one of the preliminary steps in Natural Language Processing (NLP). It is the process of assigning lexical categories to each and every word in a document based on its definition and context. POS taggers try to assign syntactic categories to words based on its role in the sentence. The requirement of a standard dataset is also necessary for developing POS taggers. Unfortunately, there is no publicly available corpus for Malayalam. In this work, we have prepared a publicly available tagged dataset as well as a CRF based POS tagger for Malayalam. The proposed system provides better results in comparison with the existing methodologies.

Index Terms—POS Tagging, Malayalam, CRF, Tagged corpus

I. INTRODUCTION

POS tagging is one of the main building blocks in many NLP applications. Resolving the ambiguities involved in the tagging process is a challenging task [1]. There are various POS taggers which differs in their internal model and amount of training. All of them can be broadly classified into three-namely rule-based, statistical and machine learning approaches. Among stochastic models, HMM is the most popular one.

Rule-based approaches use grammar rules to improve the accuracy of tagging. They consist of a two-stage architecture for POS tagging. The first stage assigns all possible POS tags to each and every word using dictionaries. The second stage makes use of dictionaries to find out the most suitable tag for each word in the document. The main bottleneck of this approach is building handcrafted rules, which is a labor-intensive and time-consuming task. On the other hand, statistical taggers make use of a tagged corpus to solve the tagging problem. They compute the probability of a given word having a specified tag in a particular context. The statistical occurrence of the tag N-gram and word-tag frequencies are used to find the most probable tag sequence for a sentence. POS tagging finds its applications in information retrieval, text to speech, information extraction and much more higher level NLP tasks such as parsing, semantics, machine translation etc. Annotated corpora are one of the main requirement for many NLP tasks such as information retrieval, machine translation, question answering etc [2]. It is a building block for constructing statistical models for natural language processing. A lot of such corpora are available for different languages across the world. Various studies on POS tagging are carried out in different languages as the required resources are available for such languages. But Malayalam is far behind as compared with such languages since it doesn’t have such useful resources.

II. ABOUT MALAYALAM AND THE CORPUS

Malayalam is one of the four major Dravidian languages with rich literary tradition. It is the official language of Kerala and Lakshadweep. Malayalam has inherited a lot from the Sanskrit, the language of Vedas. Influence of Sanskrit is evident in the alphabet, vocabulary, and phonology of Malayalam. Malayalam is a highly agglutinative language [3]. The highly productive morphology of Malayalam results in the creation of many complex and ambiguous words. Spoken forms of Malayalam is different from different parts of Kerala even though the literary dialect throughout Kerala is uniform. One of the major challenges in processing highly inflected languages like Malayalam is to deal with its frequent morphological variations of root words appearing in the text. So possible transformations through which surface words are formed from stems have to be studied in detail. In Malayalam, about 40% of words are inflected. A particular root word can form many morphological variants [4]. The high level of inflections on root words causes several problems in developing Natural Language Processing tools.

There is no benchmark corpus available for Malayalam. Earlier works are based on self-developed training and test datasets. And these datasets are not publicly available for experimentation. Hence it is not possible for the researchers in this area to compare and validate their results with the existing systems. Tagged corpus is also required in other fields of Natural Language Processing such as text to speech, information retrieval, machine translation, etc. The above-mentioned facts motivated us to develop a publicly available tagged corpus for Malayalam. So that the other researchers in this area can work on it and report their results.

As part of the dataset preparation, we have downloaded a lot of Malayalam text from available literature, science, and online newspapers. They are converted to standard UTF-8 encoding format. Useless symbols and abnormalities are removed from the text. A total of 28.7k words are prepared and tagged with
TABLE I: Tags and descriptions

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>Common noun</td>
<td>RB</td>
<td>Adverb</td>
</tr>
<tr>
<td>N</td>
<td>Proper noun</td>
<td>PSP</td>
<td>Postposition</td>
</tr>
<tr>
<td>N</td>
<td>Locative noun</td>
<td>CC</td>
<td>Co-ordinator</td>
</tr>
<tr>
<td>PR</td>
<td>Personal pronoun</td>
<td>CC</td>
<td>Subordinator</td>
</tr>
<tr>
<td>PR</td>
<td>Reflexive pronoun</td>
<td>CC</td>
<td>UT</td>
</tr>
<tr>
<td>PR</td>
<td>Relative pronoun</td>
<td>RP</td>
<td>Default particle</td>
</tr>
<tr>
<td>PR</td>
<td>Reciprocal pronoun</td>
<td>RP</td>
<td>CL</td>
</tr>
<tr>
<td>DM</td>
<td>Deictic demonstrative</td>
<td>RP</td>
<td>Intersensitive particle</td>
</tr>
<tr>
<td>DM</td>
<td>Relative demonstrative</td>
<td>RP</td>
<td>Negation particle</td>
</tr>
<tr>
<td>DM</td>
<td>Wh-word</td>
<td>QT</td>
<td>General quantifier</td>
</tr>
<tr>
<td>V</td>
<td>Main verb</td>
<td>QT</td>
<td>Cardinals</td>
</tr>
<tr>
<td>V</td>
<td>Finite verb</td>
<td>QT</td>
<td>Ordinals</td>
</tr>
<tr>
<td>V</td>
<td>Non-finite verb</td>
<td>RD</td>
<td>Foreign words</td>
</tr>
<tr>
<td>V</td>
<td>Infinite verb</td>
<td>RD</td>
<td>Symbol</td>
</tr>
<tr>
<td>V</td>
<td>Verbal noun</td>
<td>RD</td>
<td>Punctuation</td>
</tr>
<tr>
<td>V</td>
<td>Auxiliary verb</td>
<td>RD</td>
<td>Unknown</td>
</tr>
<tr>
<td>J</td>
<td>Adjective</td>
<td>RD</td>
<td>Echo words</td>
</tr>
</tbody>
</table>

Fig. 1: The percentage of most common tags extracted from training corpus

Fig. 2: Graphical representation of CRF
BIS Tags. The tagset contains 36 tags from the Bureau of Indian Standards (BIS) tag set. BIS tagset is developed by the POS tag standardization committee of Department of Information Technology (DIT), New Delhi, India. The tagset is constructed with the guidance of experts from the area of Natural Language Processing and language technology of Indian languages. Different tags and their descriptions are shown in Table 1. The tagged corpus is made publicly available at ‘www.cs.cusat.ac.in’. The percentage of most common tags extracted from the tagged corpus is shown in Figure 1.

III. RELATED WORK

POS tagging in Malayalam is still in its childhood as compared with other Dravidian languages. Only a few numbers of works are reported till now. The first work is a stochastic HMM-based methodology reported by Manju K and Soumya S in 2009 [5]. They used a very small sized training corpus for their work. Only 1400 words are used for training which was the main bottleneck of their work. The second work is reported from Amritha University Coimbatore in 2010 [6]. A tagged corpus of more than one lakh words is used for training. They have used their own tagset of 29 tags. SVM algorithm is used for training. One more work is reported in 2010 from IIT-MK Trivandrum using TNT and SVM Tool [7]. They compared the performance of TNT and SVMTool through Malayalam POS tagging using IIT-Hyderabad tagset. Another work is reported by Robert Jesaraj in 2013 [8]. A Memory-Based Language Processing (MBLP) approach is used for tagging. He utilized the power of efficient storage of solved examples and similarity-based reasoning. In 2015, a hybrid method of POS tagging was reported by Sunitha C [9]. She used handcrafted rules as well as bigrams for POS tagging. Apart from the above-mentioned studies, there is hardly any work found in the literature that directly addresses the problem of POS tagging in Malayalam, indicating a big gap in the area of Natural Language Processing.

IV. PROPOSED METHOD

Our objective is to build a POS tagger which tags all the words in the text document with BIS tags. Conditional Random Field is used for tagging purpose. CRFs are capable of predicting multiple variables that depend on each other. They are probabilistic graphical models for labeling sequential data. They try to define a conditional probability distribution over tag sequences given a particular word sequence by considering the context in to account. The graphical representation of CRF is shown in Figure 2. The main advantage of conditional random fields over HMM is their conditional nature, resulting in relaxation of independence assumptions to ensure tractable inference. CRFs are also free from label bias problem, a weakness exhibited by Maximum Entropy Markov Models.

The architecture of the system is shown in Figure 3. It contains mainly four modules. The first module is a preprocessing module that takes the tagged text as input and converts it into sequences of sentences and sequences of tags. Figure 4 shows this procedure. CRF finds the best tag sequence corresponding to a word sequence as shown in equation 1.

\[
\hat{y} = \arg \max_{\hat{y}} P(\hat{y} | \bar{w})
\]

\[
P(\hat{y} | \bar{x}, \bar{w}) = \frac{\exp(\bar{w} \cdot F(\bar{x}, \hat{y}))}{\sum_{\hat{y} \in Y} \exp(\bar{w} \cdot F(\bar{x}, \hat{y}))}
\]

\[
F(\bar{x}, \hat{y}) = \sum_{i} f(y_{i-1}, y_{i}, \bar{x}, l)
\]

Here \(\bar{x}\) is the observable word sequence and \(\hat{y}\) is the corresponding hidden tag sequence. The probability of a tag sequence, \(\hat{y}\) for a given word sequence \(\bar{x}\) is calculated as shown in equation 2. Where \(\bar{w}\) denotes the weight vector and \(F\) is the global feature vector. The global feature vector is defined by a set of local feature functions. Equation 3 elaborates this point. Each feature function can analyze the entire observation sequence \(\bar{x}\), the current \(y_{i}\) and previous \(y_{i-1}\) positions in the tag sequence and current position \(i\) in the observation sequence. A feature function is computed by summing \(f_{k}\) over all \(n\) different state transitions \(\hat{y}\). Finally decoding of the best tag sequence is done by using viterbi algorithm.

The second module in the architecture is the feature preparation module. Each word from the sentence is sent to the feature preparation module. The feature preparation module replaces each word by a set of features corresponding to that word as shown in Figure 5. The set of features we have considered for each word are suffixes of length 2, 3 and 5, the word itself, previous word, the word before the previous word, next word and the word after the next word.

The third module is the training module where the model parameters are learned. Pycrfsuite [10], a python based implementation of CRF is used for training. After training, the model is saved for testing. The last module is the testing module, where the saved model is used for testing. The words in the test data are also converted into feature vectors as in the pre-processing stage. Sequences of words are replaced with sequences of feature vectors. Then these sequences are provided to the saved CRF model for predicting the output.

V. EXPERIMENTS AND RESULTS

We have used CUSAT Malayalam corpus for our experiments. Most of the words in the corpus are unique and ambiguous which makes the tagging problem a difficult one. The most common tag in the training corpus is Noun. The preprocessed tagged text is used for training and testing. The system is trained on 23K words and tested on 5.7K words. Setting the model parameters is an important task for CRF training. Setting the model parameters is an important task for CRF training. We have set the coefficient of L1 penalty as ‘1.0’ and L2 penalty as ‘1e-3’. The model is trained for 50 epochs. The system performs with an accuracy of 91.2% on test data. An example of the tagged text tagged using CRF tagger is shown in Figure 6. The performance of different tagging algorithms on CUSAT corpus is shown in Figure 7.
VI. CONCLUSION

In this paper, we have discussed a CRF based POS tagger for Malayalam, a morphologically rich language. The odd feature of this tagger is its accuracy in tagging. Even though different taggers are available for Malayalam, CRF based
tagger outperforms all the existing methodologies. Incorporating morphological features in training helped to improve the accuracy of tagging. Out of vocabulary words are also tagged with the help of morphological and contextual features of the words. The proposed method can also be applied to various language processing applications like Named Entity Recognition, Speech Recognition, Phrase Chunking, etc.

REFERENCES


Fig. 6: An example of input and tagged output

Fig. 7: The performance of different tagging algorithms on CUSAT corpus