International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org
Volume 4, Issue 9, September 2015 **ISSN 2319 - 4847**

Hierarchical Structure Generation for Malayalam using Hybrid Approach

RAJEEV R R[1], DHANYA L K[2] , ELIZABETH SHERLY[3]

[1]IIITM-KERALA, Thiruvanathapuram

[2] CSE Dept, LBS College of Engineering Kasaragod

[3]IIITM-KERALA, Thiruvanathapuram

ABSTRACT

*In **this paper, a hierarchical structure is generated for each Malayalam sentence that contains all grammatical information including subject–object and gender information. This structure resembles to tree structure, root 'S' represents sentence and each branch represents information of each word in the sentence and leaf nodes are terminals that are only Malayalam words. This detailed syntactical structure including gender is developed and tested against a corpus of about 100 sentences (740 words), which are entered in Malayalam script.***

Keywords :- parse tree, Natural Language Processing, Parsing, hierarchical structure

1.INTRODUCTION

The computation of syntactic structure of a sentence by assigning a suitable structure is called parsing. The grammar is a formal specification of the structure allowable in the language and the parsing technique is the method of analyzing an input sentence to determine its structure according to the grammar. The major pipeline in the development of hierarchical structure generation includes tokenization, POS tagging and chunking. A hybrid approach based on rule based and statistical machine learning is proposed in this work. Malayalam is a language with rich morphology and high agglutination and is of free order. Literature shows that the rule based grammar refinement process is extremely time consuming and difficult and failed to analyee accurately a large corpus of unrestricted text. Hence, most modern parsers are based on statistical or at least partly statistical, which allows the system to gather information about the frequency with which various constructions occur in specific contexts. Any statistical approach requires the availability of aligned corpora which are: large, good-quality and representative.

2.RELATED WORK

Akshar Bharati and Rajeev Sangal developed grammar formalism, Paninian Grammar Frame- work" that has been successfully applied to all free word Indian languages. They have described a constraint based parser for the framework.

Paninian framework uses the notion of karaka relations between verbs and nouns in a sentence. Experiments show that the Paninian framework applied to modern Indian languages performs better accuracy. Selvam M and Thangarajan R have attempted to build phrase structured hybrid language model. In the development of hybrid language model, new part of speech tag set for Tamil

Language has been developed with more than 500 tags which have the wider coverage. A hybrid language model has been trained with the phrase structured Treebank using immediate head parsing technique. This paper discussed the disadvantages of CFG, as well as advantages of PCFG.There are various papers available for POS tagger for Malayalam; In Paper [3] "Parts of Speech Tagger and Chunker for Malayalam Statistical Approach"(Jisha P. Jayan and Rajeev R.R.), a statistical approach with the Hidden Markov Model following the Viterbi algorithm is described. The corpus both tagged and untagged used for training and testing the system is in the Unicode UTF-8 format. In this paper they are using at tag set for Malayalam (IIT Hyderabad Tag set).

There are some works related to Parser for Indian Languages. For Kannada language, Penn Treebank based statistical syntactic parser is developed in 2010 [1]. The Penn Treebank structure was used to create the corpus for statistical syntactic parser. The proposed syntactic parser was implemented using supervised machine learning and probabilistic context free grammars approach. Experiment shows that the performance of the proposed system is significantly good and has very competitive accuracy.

3.PROPOSED SYSTEM

The proposed system will generate a hierarchical structure while entering a Malayalam (simple/complex) sentence in Malayalam script. It is implementing using hybrid approach, include rule based as well as statistical machine learning

and probabilistic method. These approaches are Unicode-based and reduce the use of lexical dictionaries. The scope of the parser is not limited to the machine translation scenario. It can also adapt too many other NLP tasks for Malayalam language such as information extraction, relationship extraction, anaphora resolution, semantic role labelling, named entity recognition etc. In this system .The work is performed in four phases, serves as separate modules and the result of each phase connects as the input to the subsequent phase. At the end of parsing phase parse tree as well as transformation grammar with probability (PCFG) also generated.

The pipeline starts with a tokenization that splits sentences and produce output as tokens. In tokenization, rules are applied for proper identification of tokens. The POS tagger will then produce appropriate Part Of Speech (POS) tag to Tokens, and the chunking phase combine POS Tags to appropriate phrases as noun phrase, verb phrase etc

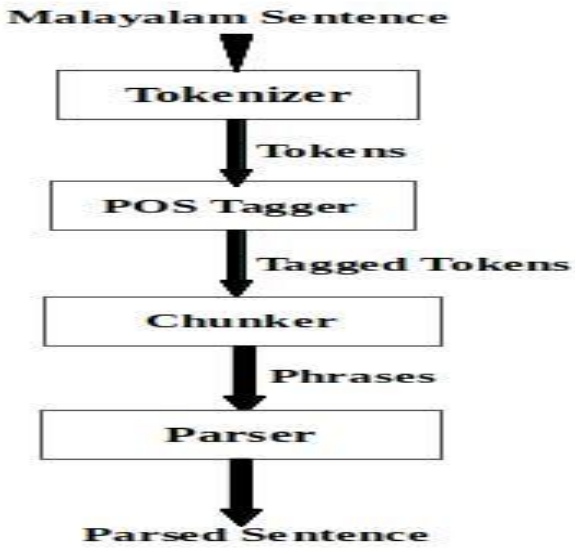

Fig1: Architecture of Proposed System

Tokenization phase uses reverse form of sandhi rules, described in Kerala Panineeyam. The tokenizer developed acts as a pre-processor for morphological analyzer and POS tagger and is implemented based on Unicode. Since a single rule solves many different word combinations, it outperforms the current systems. This component is essentially required as the first pre-processing step for languages like Malayalam and any other agglutinative languages. In the tokenization system, a supervised machine learning approach is used by which the system is able to classify the tokens into two groups: the words to be split and not to be split. Subsequently a rule-based sandhi-splitter is generated by incorporating sandhi rules. The compound words to be split are manually tagged as Compound Words (CW) and the words need not be split are tagged as Single Words (SW). Supervised machine learning approach uses Trigram n Tagger (TNT) to classify words into two groups.

Parts of Speech Tagging, a grammatical tagging, is a process of marking the words in a text as corresponding to a particular part of speech, based on its definition and context. This is the first step towards understanding any languages. The hierarchical tag set BIS tag set is used for tagging. Commonly chunking (shallow parsing) for Malayalam is implemented using statistical approach. But here rule based approach is used for implementing chunking. Nouns, Pronouns, Demonstratives, Quantifiers are grouped in to the phrase called Noun Phrase (NP). Finite verbs, Auxiliary verbs are grouped into Finite verb phrase (VGF) and Non finite verbs are grouped into Non finite Verb Phrase (VGNF) Adverbs and adjectives are grouped to for RBP and JJP respectively.

Malayalam is categorized under S-O-V order language, the default or unmarked order of constituents is Subject first, then the Object and finally the verb. But practically it is a less word order language. Hence the identification of subject and object in a sentence is very important and has many applications in natural language processing area. Hence two more levels are introduced in BIS Tag set, one for subject and object and the other for gender. Usually these modifications are needed in noun, so the modified tag set for noun is displayed in table 1. The new subtypes introduced are: Subject (S), Object (O), Male (M), Female (F), and Neutral (N).

The Chunking is the process of assigning different types of phrases in sentences. This is also known as shallow parsing. The chunking can be taken as the first step before parsing. Mostly chunking occurs after POS tagging. This is very important for activities relating to language processing. It can be used for Information Retrieval Systems, Information Extraction, Text Summarization and Bilingual Alignment. In addition, it is also used to solve computational linguistics tasks such as disambiguation problems.

Commonly chunking (shallow parsing) for Malayalam is implemented using statistical approach. But here rule based approach is used for implementing chunking.CFG, sometimes called a phrase structure grammar plays a central role in the description of natural languages. In general a CFG is a set of recursive rewriting rules called productions that are used to generate patterns of strings. The transformation grammar for the given sentence "സീത സിനിമ കണ്ടു"is given below.

S → NP VGF

NP--->N_NN_S_F N_NN_O_NU

VGF--->V_VM_VF

N_NN_S_F--->സീത

N_NN_O_NU--->സിനിമ

V_VM_VF--->കണ്ടു

(S (NP (N_NN_S_F (സീത))) (NP (N_NN_O_NU (സിനിമ))) (VGF (V_VM_VF (കണ്ടു))))

The different parts-of-speech tags and phrases associated with a sentence can be easily illustrated with the help of a syntactic structure. NLTK (Natural Language Tool Kit) is used for tree generation. There is one successful package called draw trees, by passing parsed output into this function will generate following parse tree. The parse tree for the given sentence "സീത സിനിമ കണ്ടു"is given below.

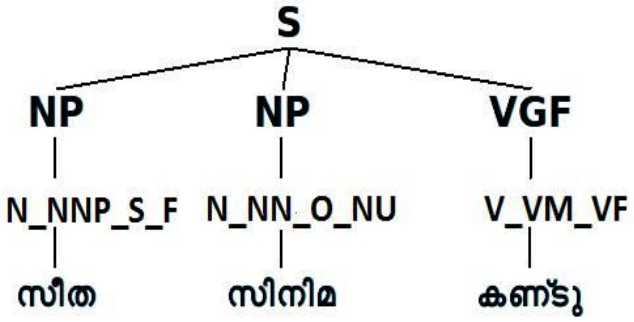

Fig 2: Final output of system

In this work, grammar was generated using rule based method and that grammar has lot of applications. By leftmost derivations system can generate sentences from transformation grammar and this will help to calculate the probability of each rule in transformation grammar of a sentence. Transformation Grammar has three levels, root level grammar, which contain 'S' and phrases. "S" will represent sentence and second level will contain productions for phrases that is actually pos tags. The last level is the productions for tags that will reveal the words. In table 1, the real necessity of transformation Grammar is explained, because it top down parser we have to generate sentence from grammar by leftmost derivation.

4.PERFORMANCE EVALUATION

The performance of the system is mainly depends on the accuracy of tagging phase, so system was evaluated using tnt-diff module and the incorrect outputs were noticed. The system performance was considerably increased by adding the input sentences.

The performance of the system was evaluated using tnt-diff module and the incorrect outputs were noticed. The system performance was considerably increased by adding the input sentences. The graph in figure 7 shows the performance of proposed parser. We trained the systems with corpus size of 2000, 3000, 4000 and 5000 sentences, which contains 62347 tokens. Then system performance was evaluated with a set of 100 distinguished sentences (740 tokens) that were out of corpus.

Overall result:

Equal: 712/ 740 (96.21%)

Different: 28/ 740 (3.78%)

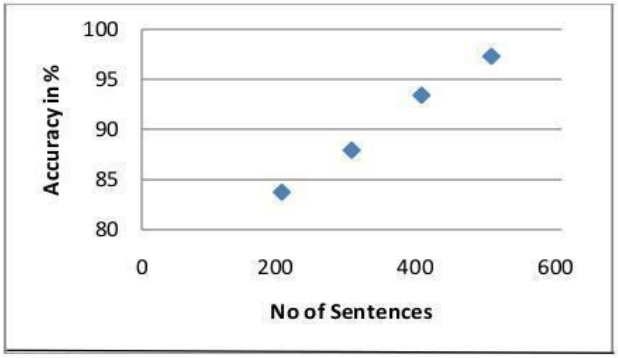

Fig 3: Performance Graph of Pos Tagging Module

We found out from the experiment that, by increasing the corpus size, the performance of our system is significantly well and achieves very competitive accuracy.

The precision for a class is the number of true positives (i.e. the number of items correctly labelled as belonging to the positive class) divided by the total number of elements labelled as belonging to the positive class (i.e. the sum of true positives and false positives, which are items incorrectly labelled as belonging to the class). Recall in this context is defined as the number of true positives divided by the total number of elements that actually belong to the positive class (i.e. the sum of true positives and false negatives, which are items which were not labelled as belonging to the positive class but should have been).In information retrieval, a perfect precision score of 1.0 means that every result retrieved by a search was relevant (but says nothing about whether all relevant documents were retrieved) whereas a perfect recall score of 1.0 means that all relevant documents were retrieved by the search (but says nothing about how many irrelevant documents were also retrieved). The evaluation of the system is done with four corpora from different domain which contains various types of words.

Corpus	Total	TP	FP	TN	FN	Precision	Recall
Corpus1	200	160	22	10	8	0.94	0.95
Corpus2	400	370	10	8	12	0.978	0.968
Corpus3	500	470	7	10	13	0.979	0.97
Corpus4	600	580	15	1	4	0.998	0.993

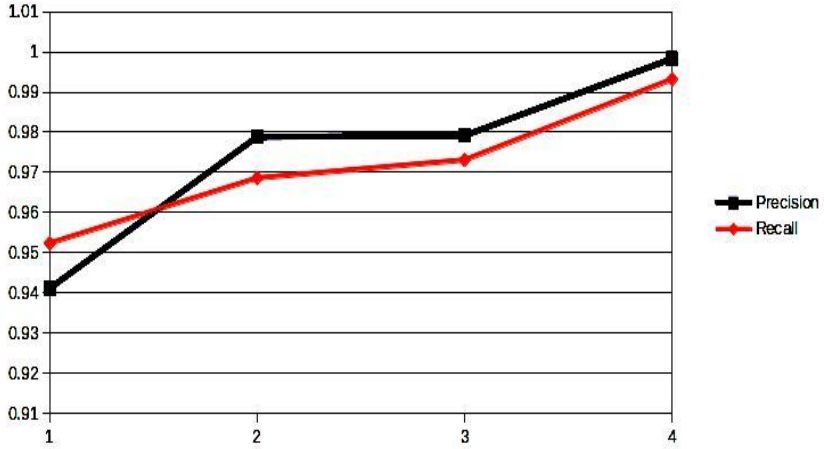

Fig 4: Performance Graph of Total system

5.CONCLUSION AND FUTUREWORK

The unavailability of large volume of corpus is a big handicap. The proposed PCFG grammar generator has been tested with 100 distinguished sentences, and the result obtained is promising and encouraging. Subsequently created Treebank corpus that may be further utilized for statistical parsing.

The accuracy of this work can be increased by expanding the size of the training corpus further. Increasing the annotated corpus size is meaningful only if there is a sufficiently large corpus of good quality available. It can also be utilized to improve many other NLP tasks such as anaphora resolution, relationship extraction, named entity recognition. Parser can be used in the future to improve these NLP tasks in Malayalam. The accuracy of probability calculation can also be increased by expanding the no of sentence.

REFERENCES

[1] Manju K, Soumya S, and Sumam Mary Idicula, "Development of a POS Tagger for Malayalam -An Experience," artcom, International Conference on advances in Recent Technologies in Communication and Computing, pp.709-713, 2009.
[2] Rajeev R R, Jisha P Jayan, and Elizabeth Sherly, "Parts of Speech Tagger for Malayalam", IJCSIT, Vol.2, No. 2, pp. 209-213, 2009.
[3] Jisha P. Jayan and Rajeev R.R. "Parts of Speech Tagger and Chunker for Malayalam Statistical Approach". Computer Engineering and Intelligent Systems 2.2, pp. 68-78 ,2011.
[4] Jurafsky D, Martin J H, "Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics and Speech Recognition", Pearson Education Series, 2002
[5] Latha R Nair,David Peter S,"Language Parsing and Syntax of Malayalam Language" , 2nd International Symposium On Computer, Communication, Control and Automation 3CA, pp. 235-238, 2013.
[6] Jisha P. Jayan, Rajeev R R, S Rajendran, Morphological Analyzer and Morphological Generator for Malayalam - Tamil Machine Translation, International Journal of Computer Applications 13.8, pp.15-18, 2011.
[7] Latha R Nair, David Peter S, Renjith P Ravindran, A System for Syntactic Structure Transfer from Malayalam to English, International Conference on Recent Advances and Future Trends in Information Technology, pp.110-122, 2012.
[8] Remya Rajan, Remya Sivan, Remya Ravindran, K.P Soman. Rule based machine translation from english to malayalam. In Conference Proceedings on International Conference on Advances in Computing, Control, and Telecommunication Technologies, pp. 439-441, 2009.
[9] Mary Priya Sebastian, G Santhosh Kumar, English to malayalam translation: a statistical approach. In Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India, page 64. ACM, 2010.