International Journal of Engineering Research and Development
e-ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com
Volume 9, Issue 3 (December 2013), PP. 28-33

Semantic Role Extraction and General Concept Understanding in Malayalam using Paninian Grammar

Radhika K. T.[1], Dr.P.C. Reghu Raj[2]
Dept. of CSE, Govt. Engg. College Sreekrishnapuram, Palakkad, Kerala.

Abstract: - The collection of methods by which human languages convey meaning is called meaning structure of a language. It includes many conventional form-meaning associations, word-order regularities, tense systems, conjunctions and quantifiers, and a fundamental predicate-argument structure. In the Dravidian language, Malayalam, the Karaka theory, is useful for both the syntax analysis and semantic analysis of Malayalam sentences. Here proposes a system that builds semantic structure from a given Malayalam text using semantic roles (karakas) extraction. Semantic structure can be utilized for solving many language computing tasks such as Conceptual Indexing, Conceptual searching and retrieval, automatic question answering, automatic text summarization, syntactic and semantic analysis of Malayalam documents, etc.

Keywords: - Natural Language Processing**,** Malayalam, Vibhakthi, Karaka, POS tagging, compound words, Semantic structure.

I. INTRODUCTION

The goal of the Paninian approach is to construct a theory of human natural language communication. Grammar as part of such a theory of communication, is a system of rules that establishes a relation between what the speaker decides to say and his utterance, and similarly what the hearer hears and the meaning he extracts [1]. Every verbal root (dhathu) denotes an action consists of an activity and a result.

Result is the state which when reached, the action is complete. Activity consists of actions carried out by different participants or karakas involved in the action. Parse structure for a sentence consists primarily of the verbal groups and the nominal in the sentence, and the karaka relations among them as in Fig 1. Karakas provides maximum necessary information relative to a verb.

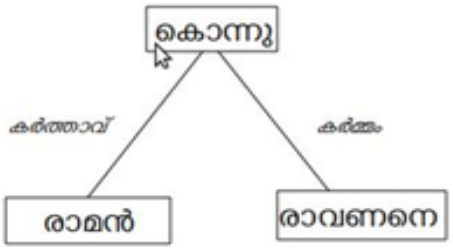

Fig.1: Nouns as arguments of verb

II. VIBHAKTHI IN MALAYALAM LANGUAGE

The study of roles associated with specific verbs and across classes of verbs is called thematic role analysis or case role (karaka) analysis. In Malayalam, the root word gets inflected to change its meaning, or to relate the word with other words. For example take the words 'രാമൻ' and 'പശു'. There is no meaning in the sentence 'രാമൻ പശു'. When we add morpheme 'ന്റെ' to the word 'രാമൻ', we get a meaning from the sentence 'രാമന്റെ പശു' . The morphemes which are added at the end of a word to make it more meaningful and to relate them with other words are called prathyayas. Prathyayas are added to nouns for mainly three purposes:

1) Linga Prathyayam(To change Gender) Example: കേമൻ(അൻ), കേമി(ഇ)
2) Vachana Prathyayam(To change Number) Example: അമ്മമാർ
3) Vachana Prathyayam(To change Number) Examples:

4) Vibhakthi Prathyayam((To relate nouns with other words) Example: 'രാമന്റെ പശു'.

According to A.R.Raja Raja Varma Malayalam have seven vibhathies [3] as shown in Table 1.

TABLE I

VIBHAKTIES OF MALAYALAM

വിഭക്തിയുടെ പേര്	പ്രത്യയം	ഉദാഹരണം
നിർദ്ദേശിക	പ്രത്യയമില്ല	കുട്ടികൾ
പ്രതിഗ്രാഹിക	എ	മനുഷ്യരെ
സംയോജിക	ഓട്	മനുഷ്യരോട്
ഉദ്ദേശിക	ഉ്,ക്ക്	രാമന്, സീതയ്ക്ക്
പ്രയോജിക	ആൽ	മനുഷ്യരാൽ
സംബന്ധിക	ന്റെ, ഉടെ	മനുഷ്യരുടെ
ആധാരിക	ഇൽ, കൽ	മനുഷ്യരിൽ

III. RELATED WORK

In Malayalam, the root word gets inflected to form new words. In addition to this, two or more words combined to form a single word based on set of rules called 'sandhi rules'. The article [10] investigates the rules of inflection and agglutination and explains how it is challenging to some computer based language processing. Another work describe a process for developing a parser by incorporating the Paninian grammar concepts in Malayalam [2]. In this paper they have shown how computational Paninian grammar can be used for parsing Malayalam language. The parser is based on translating grammatical constraints to integer programming constraints.

Another work presents a parser for simple Malayalam sentences. It uses a grammar checker for parsing simple Malay-alam sentences [11].

IV. PROPOSED SYSTEM

The proposed system architecture is shown in Fig.2. Its important modules are explained below.

A. Tokenizer

In order to understand the concepts from text, it is first necessary to interpret the text as a sequence of words, occasionally interrupted by punctuation marks. This is the job of the tokenizer, which segments a character stream into a sequence of tokens. The goal of the tokenizer is to group characters into units that can be looked up in a lexicon to determine what is known about them and how to treat them. Tokenization of Malayalam text can be done in the same way as English language tokenization.

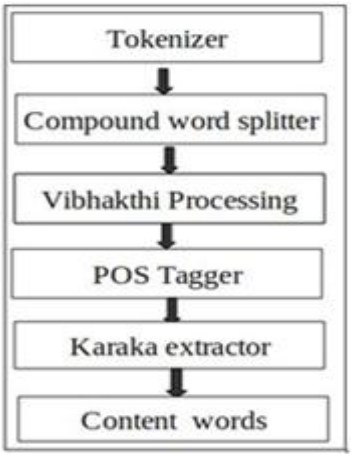

Fig. 2. Stages involved in content word extraction

B. Compound word splitter

Splitting compound word into appropriate forms is an important task in Natural language Processing (NLP) applications. Splitting is needed for compound words whose morphemes are of different lexical categories. Example: 'കുട്ടിയെഴുതി'. A compound word generally consists of noun-noun, noun-adjective, verb-noun, adverb-verb and adjective-noun combinations. In some cases all the words of an entire sentence may

combine to form a single one.
Hybrid sandhi-splitter is a program which is developed to split such compound words into its constituent morphemes; compound words are split according to the reversed form of sandhi rules proposed in Keralapanineeyam [3]. This module mainly consists of two phases: statistical learning phase (su-pervised) and a splitting module. Block diagam is shown as Fig 3. [12].
1) Supervised Learning Module: Before splitting, both the single words and compound words are recognized through a supervised machine learning approach using TnT tagger. TnT, the short form of Trigrams 'n' Tags, is a very efficient statistical tagger that is trainable on different languages andvirtually any tag set. The compound words to be split are manually tagged as < SPT > and the words need not be split are tagged as < SPF >. This Specially tagged (labeled instances) data is supplied to the TnT tagger for learning [12]. Training data supplied to the TnT tagger must be in a two-column file format. After training, it will build a model of training data. This model can be later used to tag a new file. The input file contains one token per line. In the basic mode, the tagger adds a second column to each line, containing the tag for the word.

2) Splitting Module: Rule- based Splitter is the second component of compound word splitter. It contains a set of reversed sandhi rules. Reverse computation of sandhi means application of Paninian rules in reverse form for splitting. As mentioned earlier, sandhi can take place between vowel and vowel, vowel and semivowel, semivowel and semivowel, consonants and consonants and between visarga and other sounds.

Fig. 3. Block diagram of compound word splitter

IV. VIBHAKTHIS AND KARAKAS

During the semantic analysis, verb is taking as the central, binding element of the sentence. Sanskrit grammarians like Panini had used this idea in their grammar. The relation between noun and verb in a Malayalam sentence is called karakam. As illustrated in Table1, Sambadhika vibhakthi shows relationship with noun. All vibhakthi prathyayas except sambadhika indicates karakas. So vibhakthi prathyayas are also called karaka prathyayas. The proposed system makes use of this relation between vibhakti and karaka roles in Malayalam sentences. There are 7 karakas.

(i) Karthru karakam –Subject
(ii) Karma karakam- Object
(iii) Karana karakam- Instrument
(iv) Kaarana(Hethu) karakam- Instrument
(v) Sakshi karakam- Experiencer
(vi) Swami karakam- Beneficiary
(vii) Adhikarana karkam- Locative

A. Vibhakthi to Karaka mapping

As illustrated in Table I, case endings differentiate the vibhakthis. After finding vibhakthis from the tokens of the given text, karakas are found out by mapping as specified in the Table II .Mapping between vibhakthi prathyayas and karakas is shown in Table II.

In addition to this vibhakthis, karaka rules are extended with *ഗതികൾ* like *കൊണ്ട് പറ്റി* etc. Also there are separate rules to understand and process the roles of *നപുംസകങ്ങൾ* (neutal gender). Eg: 'ദേവൻ വെള്ളം കുടിച്ചു' ' here the word വെള്ളം is a neutral gender. According to the rule, that word is treated as 'വെള്ളത്തെ' and its semantic role is കർമ്മ കാരകം (object). Fig 4 illustrates result of karaka extraction for the sentence 'സീതയുടെ പിതാവ് ജനകനാണ്'.

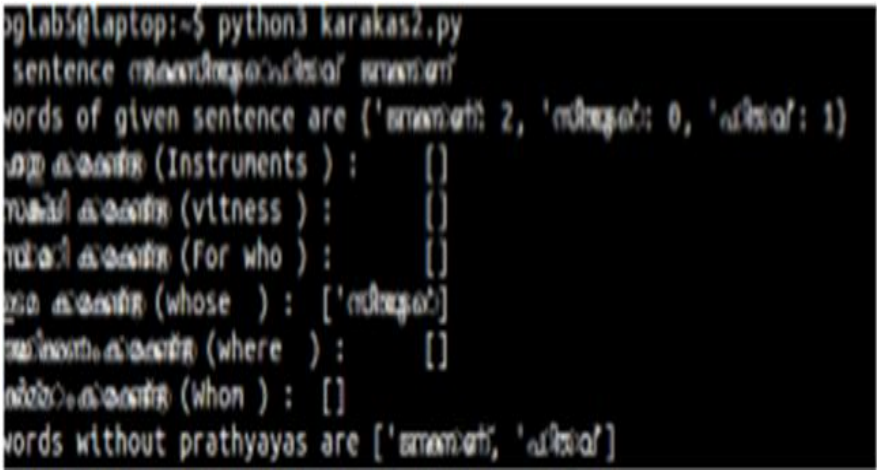

Fig. 4. Extraction of karakas from a sentence

VI. POS TAGGING USING TNT

After finding karakas, POS tagging was applied to the remaining tokens to find Karthru karakas(nouns without prathyayas), phrases and verbs.

Tagging was doneusing a statistical approach with the help of TnT (Trigram's'N Tags). Training data was manually prepared based on BIS tagset(Bureau of Indian Standard tagset. BIS is a hierarchical tagset which uses grammatical categories and their sub categories along with other morpho-syntactic attributes. They are structured relative to one other, instead of using large number of independent labels and contains a small number of categories at the top level, with a number of sub categories in the form of a tree. Some of the BIS tags are given as :

Common Noun(NN), Proper Noun(NNP), Nloc(NST), Pro-noun(PR), Personal(PRP),Main verb VM , Finite verb(VF), Non-finite verb (VNF), Infinitive verb (VINF), Adjective(JJ), Adverb(RB) [8].

Under the assumption -semantics or concept of Malayalam text is mainly depends on verbs and nouns, proposed system takes only verbs and nouns from the input text for computation. Other tags like adjectives, adverbs etc are not included in the list of content words.

Another important property of Malayalam is the usage of phrases in text. Phrases are detected and separated using the module named phrase detection and is explained below.

VII. PHRASE DETECTION

Since phrases have a different meaning from its individual word meanings, detecting them is difficult but necessary for semantic analysis. Here phrases with two words are detected. Examples are 'പ്രകൃതി ഭംഗി', ' പ്രകൃതി ദുരന്തം'. A list of commonly found phrases having length two was collected and bigrams of the text were matched against this list for detecting phrases.

VIII. UNDERSTANDING THE GENERAL CONCEPT

As explained earlier, each karaka has a specific role in a sentence. But concept of a sentence mainly depends on Karthru karakas. Credits were assigned to each karakas in order to understand the general concept or theme of the text. Karthru karaka, swami karaka and karmma karaka get a higher credit than other karakas. Assigned credit increases as they occur frequently in a text. Term frequency was computed for this. Since sambadhika vibhakthi gives the relationship between two nouns, it is also added to the list of important words. These words are mapped to a concept dictionary to understand the general concept onto which these keywords belong to. Example: Content words like മഹാഭാരതം, വ്യാസൻ , പർവ്വങ്ങൾ etc. are mapped to the concept ഇതിഹാസകൃതി. After understanding the concept, the proposed system converts the given text to a representation called conceptual structures.

IX. CONCEPTUAL STRUCTURES

One of the simplest ways to encode the kind of properties that we have in mind is through the use of feature structures. These are simply sets of feature-value pairs, where features are unanalysable atomic symbols drawn from some finite set. It is illustrated in Fig 5. In matrix-like diagram, called a attribute-value-matrix (AVM).

$$\begin{bmatrix} \text{FEATURE}_1 & \text{VALUE}_1 \\ \text{FEATURE}_2 & \text{VALUE}_2 \\ \vdots & \\ \text{FEATURE}_n & \text{VALUE}_n \end{bmatrix}$$

Fig. 5. Feature structure

X. TESTING AND RESULTS

The karakas along with concept are arranged in the form of concept structures. Fig 6.shows the concept structure of Malayalam sentence 'പെട്രോൾ വില ലിറ്ററിന് രണ്ടര രൂപ വരെ കുറഞ്ഞേക്കും' obtained.

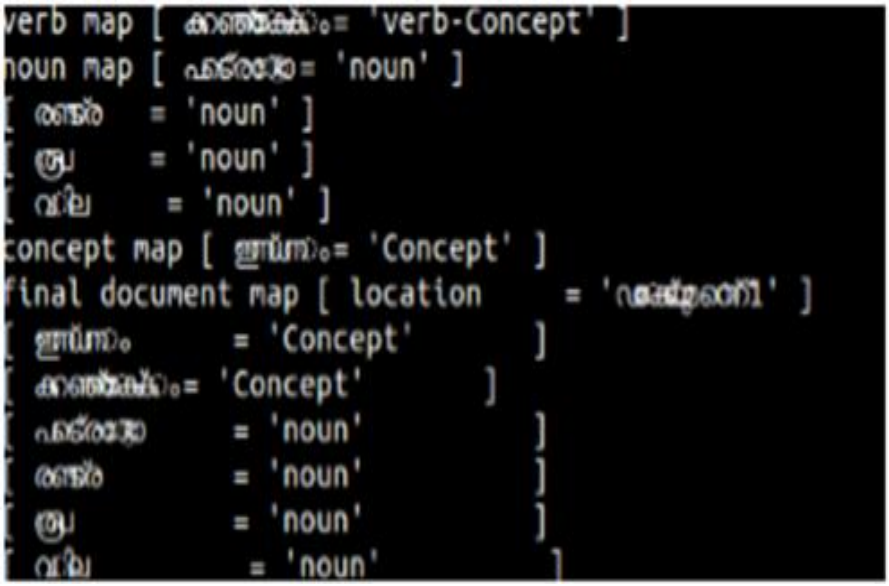

Fig. 6. Conceptual structure

In the proposed work, Karaka extraction was tested using 1000 sentences with 90 percentage accuracy. Since the concept dictionary- the back bone of the proposed system is a domain specific one, the system is able to find concept of a document within this specific domain. The result can be improved by enlarging the concept dictionary.

XI. CONCLUSION

The proposed work has importance in the Malayalam Computing area. POS tagger as well as the compound words splitter developed as part of this can be used for other NLP tasks in Malayalam. The system can be further utilized for the applications like answering questions from an essay, document summarization and natural language generation. Karaka extractor can be integrated as a module for developing a parser.

REFERENCES

[1]. Akshar Bharati, Vineeth Chaithanya, Rajeev Sangal, "Natural Language Processing: A Paninian Perspective", Prentice-Hall of India, New Delhi.

[2]. Aparnna T, Raji P G, Soman K P, "Integer Linear Programming Approach to Dependency Parsing for MALAYALAM", International Conference on Recent Trends in Information, Telecommunication and Computing,2010.

[3]. A. R. Rajaraja Varma , Keralapanineeyam, Sahitya Pravarthaka , C S Ltd., Kottayam,1968

[4]. Anish A, "Part of speech tagging for Malayalam", Amritha School of Engineering, 2008.

[5]. Daniel Jurasfky and James H. Martin, An introduction to Natural Lan-guage Processing, Computational Linguistics, and Speech Recognition, Pearson Education, Inc. 2000.

[6]. Latha R. Nair S. David Peter, Development of a Rule Based Learning System for Splitting Compound Words in Malayalam Language, IEEE journal on computational intelligence, pp-751, 2011.

[7]. Manu Madhavan, P. C. Reghu Raj "Application of Karaka Relations in Natural Language Generation", in Proc. of National Conference on Indian Language Computing, CUSAT, Kerala,2012.

[8]. Jisha P. Jayan and Rajeev R. R. "Parts of Speech Tagger for Malayalam". IJCSIT International Journal of Computer Science and Information Tech-nology. Vol.2, No.2, December 2009, pp.209-213.

[9]. Saranya S. Morphological Analyzer for Malayalam verbs, M.Tech Thesis,Amrita School of Engineering, Coimbatore,2008.

[10]. [10] Santhosh Thottingal, "Inflection and Agglutination: Challenges to Malayalam Computing"

[11]. Jasine Babu, "A parser for simple Malayalam sentences",2005.

[12]. Divya Das, Radhika K.T., Rajeev R. R, Reghu Raj P. C. Hybrid

[13]. Sandhi Splitter for Malayalam using Unicode , In Proceedings of National Seminar on Relevance of Malayalam in the Field of Information Technology, Kerala University ,2012.