International Journal of Latest Trends in Engineering and Technology
Special Issue SACAIM 2016, pp. 13-18
e-ISSN:2278-621X

KNOWLEDGE GRAPH OF MALAYALAM NOUN

Raji Sukumar A[1], Dr. Babu Anto P[2]

Abstract-Knowledge representation is a central topic in AI. Problem solving, task describing, expressing experience knowledge, inferring and decision making, all of these are based on knowledge. It has important influence in fields like pattern recognition, natural language understanding, machine study, automatic theorem proving, and expert system. Although now there are many methods for knowledge representation, such as production rules, logic, semantic network, frame. Thus, to explore new methods for knowledge representations are still identified. This paper proposes a Knowledge graph model of Malayalam noun.

Keywords –AI, KB, NLP,KG.

I. INTRODUCTION

Knowledge graphs, as a kind of representation for NLP, points out a new way for NL describing and modeling leads to semantic understanding. A concept is a component of human thought, and is the thinking unit that refers to objective things and their peculiar properties. The mind's operation of forming concepts, with the meaning of words and phrases has realized. The meanings of a word are determined by the perception of reality, which belongs to both the category of thought and language. Therefore, concept has a communication with the meaning of a word. The key of NLP is to handle the meaning of a word.

II. BACKGROUND STUDY

Like most AI systems, NLP, requires substantial amount of knowledge that is difficult to acquire. A measure of semantic similarity is presented in taxonomy based on the notion of shared information context. Systems have been developed, mainly for languagessuch as English; some examples of these systems are California Restaurant Query, ExpediaHotels, GeoQuery[1], JobQuery[2], SQ-HAL [3], andSystemX[4].NLP can play a role in both the retrieval and storage of documents which can be used to build a friendly user interface that allows free language query submission and hence eliminates the need for mastering a formal query format.The different development stages of NLP, emphasis on MT by the influence of AI. This study briefly reviews some of these techniques. Between these technologies we should mention the Semantic Technologies as part of AI, Rule-based systems, logic-based inference and decision support systems [5].Hoede and Li [7] wrote a paper on a first set of words; verbs, nouns and prepositions. Hoede and Liu [8] wrote a paper on a second set of words; adverbs, adjectives and Chinese classifiers or quantity words. Hoede and Zhang [6] wrote a paper on a third set of words.

III. PROPOSED SYSTEM

As Malayalam is an inflectionally rich language. Words of Malayalam have to be classified into different categories by defining them with different sets of orthographic rules. The linguistic aspects of orthographic rules are perceived in the previous section. The computational characteristics of the orthographic rules are discoursed in this section. The rule notation of Chomsky and Halle is followed for

[1] *Department of Information technology Kannur University, Kannur, Kerala, India*
[2] *Department of Information technology Kannur University, Kannur, Kerala, India*

Malayalam orthographic rules that results from the plural marking by morphotactics of Malayalam. Therefore six categories are sufficient for marking each inflection of Malayalam nouns. The order of orthographic rules is important; the special rules have to be considered before the general rules written. The exceptions are taken as special rules as they don't follow the general rules. Same strategies of FST that are applied to Malayalam nouns are also applicable to Malayalam verbs. FST model is built for Malayalam Verbs too. The categories considered for Malayalam verbs are different from the groups of Malayalam nouns. In Malayalam nouns, the total of 35 stem ends considered is grouped into 7 case categories for root form and 7 case categories for plural in order to optimize the rules of 15 inflections.The various rules for Malayalam Nounis given below.

Rule 1:(Nominative case marker)-The same noun word itself.
$amma + NOM \rightarrow amma$
Rule 2:(Accusative)- The accusative case marker of Malayalam is 'e'.
end with n,L, then accusative case is added to the stems without any change at the end.

$$rAman + ACC \rightarrow rAman + e \rightarrow rAmane$$
$$avaL + ACC \rightarrow avaL + e \rightarrow avaLe$$

If end with R then delete R and add r+ACC at the end.

$$avaR + ACC \rightarrow avaR + r + e \rightarrow avare$$

If end with m then delete m and add tt+ACC at the end.

$$mAnaM + ACC \rightarrow mAnaM + tt + e \rightarrow mAnatte$$

If end with a, i then add y+ACC at the end.
$kavi + ACC \rightarrow kavi + y + e \rightarrow kaviye$
If end with N or l then add in+ACC at the end.

$kooN + ACC \rightarrow kooN + in + e \rightarrow kooNine$
If end with u then add v+ACC at the end.
$kuru + ACC \rightarrow kuru + vin + e \rightarrow kuruvine$

If the word is njan then enne.njAn → enne
If the word is ni then nine. $ni \rightarrow nine$
Rule 3: The sociative case marker is 'OT'.$rAmaN + SOC \rightarrow rAmaN + OT \rightarrow rAmaNOT$

If end with N,L then add SOC at the end.$rAmaN + SOC \rightarrow rAmaN + OT \rightarrow rAmaNOT$
If end with M delete M and add ttin+SOC at the end.

$$mAnaM + SOC \rightarrow mAnaM + ttin + OT \rightarrow mAnattinOT$$

If end with i then add y +SOC at the end.
$kavi + SOC \rightarrow kavi + y + OT \rightarrow kaviyOT$

If end with R then delete R and add r+ SOC at the end.

$$avaR + SOC \rightarrow avaR + r + OT \rightarrow avarOT$$

If end with l,N then add in+ SOC at the end.
kooN+SOC→kooN+in+OT→kooNinOTu
if end with u then add vin SOC at the end.
$kuru + SOC \rightarrow kuru + vin + OT \rightarrow kuruvinOT$

If the word is njAn then ennOT.
njAn→ennOT
If the word is ni then ninnOT
ni→ ninnOT

Rule 4 : The Dative case marker for Malayalam is 'kk', which alternates with 'in'
If end with n then add u at the *end*

$$rAman + DAT \rightarrow rAman + u \rightarrow rAmanu$$

if end with L,I,R then add DAT(kk)at the end.
Eg:$avaL + DAT \rightarrow avaL + kk \rightarrow avaLkk$
If end with m then delete m and add tt+DAT(in) at the end

$$mAnam + DAT \rightarrow mAnam + tt + in \rightarrow mAnattin$$

If end with N,l then add DAT(in) at the end.

$$kooN + DAT \rightarrow kooN + in \rightarrow kooNin$$

If end with u then add v+DAT(in) at the end.
$kuru + DAT \rightarrow kuru + v + in \rightarrow kuruvin$

If the word is njAn then enikk.
$njAn \rightarrow enikk$
If the word is ni then ninakk.
$Ni \rightarrow ninakk$
Rule 5 : The Instrumental case marker is 'Al'.
If end with n,L then add INS at the end.

$$avaL + INS \rightarrow avaL + AL \rightarrow avaLAL$$

If end with m then delete m and add tt+ INS at the end.

$$mAnam + INS \rightarrow mAna\cancel{m} + tt + AL \rightarrow mAnattAL$$

If end with i then add y+ INS at the end.
$kavi + INS \rightarrow kavi + y + AL \rightarrow kaviyAL$
If end with R then delete R and add r+ INS at the end.
$avaR + INS \rightarrow ava\cancel{R} + r + AL \rightarrow avarAL$
If end with N,l then add in+ INS at the end.
$kooN + INS \rightarrow kooN + in + AL \rightarrow kooNinAL$
If end with u then add vin+ INS at the end.
$kuru + INS \rightarrow kuru + vin + AL \rightarrow kuruvinAL$
If the word is njAn then ennAl
$njAn \rightarrow ennAl$
If the word is ni then ninnAl.
$ni \rightarrow ninnAl$
Rule 6 : The genitive case marker is 'uTe'. 'Re' occurs after nominal bases or obliquebases ending in 'n', where 'uTe' occurs elsewhere
If end with n then add te at the end.
If end with L then add uTe at the end.
$avaL + GEN \rightarrow avaL + uTe \rightarrow avaLuTe$
If end with m then delete m and add tti+GEN(nTe) at the end.
$rAman + GEN \rightarrow rAman + nTe \rightarrow rAmanTe$
If end with i then add y+ GEN (uTe) at the end.
$kavi + GEN \rightarrow kavi + y + uTe \rightarrow kaviyuTe$
If end with R then delete R and add r+ GEN (uTe) at the end.
$avaR + GEN \rightarrow ava\cancel{R} + r + uTe \rightarrow avaruTe$
If end with N,l then add in+ GEN (Te) at the end.
$kooN + GEN \rightarrow kooN + in + Te \rightarrow kooNinTe$
If end with u then add vin+ GEN (Te) at the end.
$kuru + GEN \rightarrow kuru + vin + Te \rightarrow kuruvinTe$
If the word is njAn then ente
njAn → enTe
If the word is ni then ninte
$ni \rightarrow ninTe$
Rule 7 : The Locative case marker of Malayalam is 'il'.
If end with L,l,n,N then add LOC at the end.

$$avaL + LOC \rightarrow avaL + il \rightarrow avaLil$$

If end with m then delete m and add tt+ LOC at the end.

$$mAnam + LOC \rightarrow mAna\cancel{m} + tt + il \rightarrow mAnattil$$

If end with i then add y+ LOC at the end.

$$kavi + LOC \rightarrow kavi + y + il \rightarrow kaviyil$$

If end with R then delete R and add ril at the end.

$$avaR + LOC \rightarrow avaR + r + il \rightarrow avaril$$

If end with u then add v+ LOC at the end.

Eg:$kuru + LOC \rightarrow kuru + v + il \rightarrow kuruvil$

If the word is njAn then ennil.

$njAn \rightarrow ennil$

If the word is ni then ninnil.

$ni \rightarrow ninnil$

The various rules for Malayalam Number

(Masculine Gender)

Rule 1: If end with n then add mAR at the end.

$rAman + PL \rightarrow rAman + mAR \rightarrow rAmanmAR$

(Feminine gender)

Rule 2 : If end with a then add mARat the end.

$Amma + PL \rightarrow amma + mAR \rightarrow ammamAR$

Rule 3: If end with i then add kaL at the end.

$mitukki + PL \rightarrow mitukki + kaL \rightarrow mitukkikaL$

(Neuter gender)

Rule 4 : If end with a then add kaL at the end.

$ila + PL \rightarrow ila + kaL \rightarrow ila + kaL$

Rule 5 : If end with u then add KaL at the end.

$karu + PL \rightarrow karu + KaL \rightarrow karuKaL$

Rule 6 : If followed by any number does not make any change.

$anJupEna$

(Commen gender)

Rule 7 : If end with n then replace n with R.

$sambaNaN + PL \rightarrow sambaNa\cancel{N} \rightarrow sambaNa$

$ila + PL \rightarrow ila + kaL \rightarrow ila + kaL$

Rule 5 : If end with u then add KaL at the end.

$karu + PL \rightarrow karu + KaL \rightarrow karuKaL$

Rule 6 : If followed by any number does not make any change.

$$anJupEna$$

(Commen gender)

Rule 7 : If end with n then replace n with R.

$$sambaNaN + PL \rightarrow sambaNa\cancel{N} \rightarrow sam$$

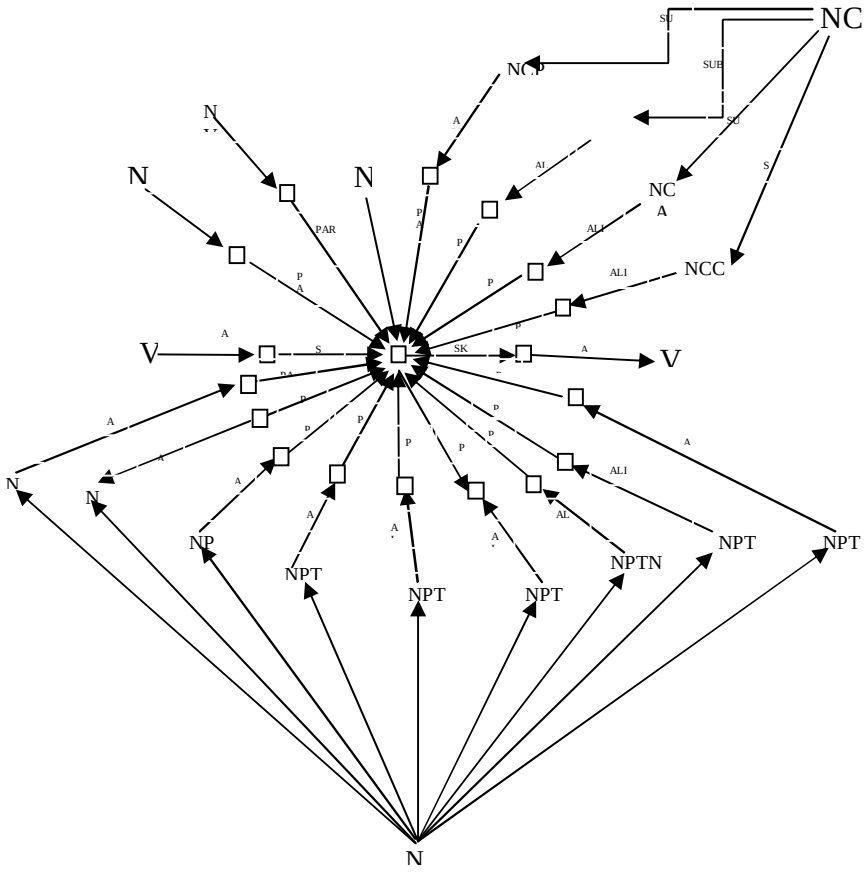
IV. Knowledge Graph of Malayalam noun

The basic theory of knowledge graphs is outlined from the point of view of ontology contrast with logic. Knowledge graphs are more general and more original than conceptual graphs because the number of its relation types is very limited, so structural parsing is introduced based on the theory of knowledge graphs. Under consideration of the semantic and syntactic features of NL, both semantic and syntactic word graphs are designed and grammar rules are derived from the syntactic word graphs and a parse tree can then be given for a sentence. The parse tree can be used to map the sentence on a sentence graph, which is called structural parsing. The relationship with utterance paths and a result chunk indicator are proposed to guide structural parsing with the problem of uttering a sentence graph. The word graph

ontology consists, eight types of binary relationships and four types of n-array relationships are also called frame relationships.

The eight binary types describe to form a knowledge graph are:

Equality	: EQU
Subset relationship	: SUB
Similarity of sets, alikeness	: ALI
Disparateness	: DIS
Causality	: CAU
Ordering	: ORD
Attribution	: PAR
Informational dependency	: SKO



N- Noun	NP- Pronoun	NCA - Abstract Noun	NCCO – Collective Noun
V- Verb	NCP- Proper Noun	NCC - Common Noun	NPTNN - NanarthakaSarvakam
NQ- Abstract Noun	NPF - First Person	NPS - Second Person	NPTNC -ChodyaSarvanamam
NV- Verbal Noun	NPT - Third Person	NC-Concrete Noun	NPTNY - VyakshepakaSarvanamam
NPTNN - NanarthakaSarvakam	NPTNS - SarvavachiSarvanamam	VPVM- Munvinayecham	NPTNR – NirdhishtavachiSarvanamam
NPTNM - AmsavachiSarvanamam			

V. CONCLUSION

Partial structural parsing is the mapping of a sentence that is in the input text onto a set of semantic chunk graphs of this sentence. The goal of partial structural parsing is creating the scenario patterns of information to be extracted. Mapping of a word on a knowledge graph, for the structural parsing, which combines the various bigger semantic chunk graphs into a sentence graph which will be useful for IE from the text input. This paper proposes the clear picture of linguistics variations of a Malayalam noun through a knowledge graph which can be useful for the knowledge extraction of a sentence.

REFERENCES

[1] Zelle J M, R J Mooney. "Learning to parse database queries using inductive logic programming. In: Proc." Thirteenth National Conference on Aritificial Intelligence, Portland (1996): 1050–1055.

[2] Thompson C A, R J Mooney, L R Tang. "Learning to parse natural language database queries into logical form." In: Proc. ML-97 Workshop on Automata Induction, Grammatical Inference, and Language Acquisition (1997).

[3] Flores V, J M, J M Matadamas H. "Computational Linguistics Laboratory: Project Sylvia-NQL." In: Proc. 7th. International Congress on Computer Science Research, Technological Institute of Cd. Madero, Tampico, Mexico (2000): 73–81.

[4] Cercone N. "Human-Computer Interfaces: DBLEARN and SystemX." International Workshop On Rough Sets And Knowledge Discovery (RSKD-93) (1993): 27-28.

[5] Locke W N and Booth A D . "Machine Translation of Languages." John Wiley (1955).

[6] Hoede and L. Zhang, Word Graphs: The Third Set. : Broadening the Base, Proc. of the 9 th International Conf. on Conceptual Structures (Eds. H. S. Delugach and G. Stumme), CA, USA, Lecture Notes in Artificial Intelligence no.2120, 15-28, 2001

[7] C. Hoede and X. Li, Word Graphs: The First Set . In: Conceptual Structures: Knowledge Representation as Interlingua, Aux. Proc. of the Fourth International Conf. on Conceptual Structures (Eds. P. W. Eklund, G. Ellis and G. Mann), Bondi Beach, Sydney, Australia, 81-93, 1996.

[8] C. Hoede and X. Liu, Word Graphs: The Second Set In:Conceptual Structures In: Conceptual Structures: Theory, Tools and Applications, Proc. of the 6th International Conf. on Conceptual Structures (Eds. M-L. Mugnier and M. Chein), Montpellier, Springer Lecture Notes in Artificial Intelligence no.1453, 375-389, 1998.