KNOWLEDGE GRAPH OF MALAYALAM NOUN

Raji Sukumar A1, Dr. Babu Anto P2

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 Nouns are given below.

**Rule 1:** (Nominative case marker) - The same noun word itself.

\[ amma \rightarrow NOM \rightarrow amma \]

**Rule 2:** (Accusative) - The accusative case marker of Malayalam is 'e'.

If 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 I 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.

\[ kuru + ACC \rightarrow kuru + vin + e \rightarrow kurwine \]

If the word is nj then enne, njAn \rightarrow 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 tin+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.

\[ kooN + SOC \rightarrow kooN + in + OT \rightarrow kooNinOTu \]

If end with u then add vin SOC at the end.

\[ kuru + SOC \rightarrow kuru + vin + OT \rightarrow kurwvinOT \]

If the word is njAn then ennOT.

njAn \rightarrow ennOT

If the word is ni then ninnOT

ni \rightarrow 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.
if end with L,LR then add DAT(kk) at the end.
Eg: aval + DAT → aval + kk → avalkk
If end with m then delete m and add tt+DAT(in) at the end
mAnam + DAT → mAnam + tt + in → mAnattin
If end with N,l then add DAT(in) at the end.
kooN + DAT → kooN + in → kooNin
If end with u then add v+DAT(in) at the end.
kuru + DAT → kuru + v + in → kuruvin

If the word is njAn then enikk.
njAn → enikk
If the word is ni then ninakk.
ni → ninakk

Rule 5 : The Instrumental case marker is ‘Al’.
If end with n,L then add INS at the end.

aval + INS → aval + AL → avalAL
If end with m then delete m and add tt+INS at the end.

mAnam + INS → mAnam + tt + AL → mAnattAL
If end with i then add y+ INS at the end.

kavi + INS → kavi + y + AL → kaviyAL
If end with R then delete R and add r+ INS at the end.
avuR + INS → avuR + r + AL → avuAL
If end with N,l then add in+ INS at the end.
kooN + INS → kooN + in + AL → kooNinAL
If end with u then add vin+ INS at the end.
kuru + INS → kuru + vin + AL → kuruvinAL
If the word is njAn then ennAl

njAn → ennAl
If the word is ni then ninnAl.
ni → ninnAl

Rule 6 : The genitive case marker is ‘uTe’. ‘Re’ occurs after nominal bases or oblique bases 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 → aval + uTe → avaluTe
If end with m then delete m and add tt+GEN(nTe) at the end.

rAman + GEN → rAman + nTe → rAmanTe
If end with i then add y+ GEN (uTe) at the end.

kavi + GEN → kavi + y + uTe → kaviyuTe
If end with R then delete R and add r+ GEN (uTe) at the end.
avuR + GEN → avuR + r + uTe → avaruTe
If end with N,l then add in+ GEN (Te) at the end.
kooN + GEN → kooN + in + Te → kooNinTe
If end with u then add vin+ GEN (Te) at the end.
kuru + GEN → kuru + vin + Te → kuruvinTe
If the word is njAn then enTe

njAn → enTe
If the word is ni then ninte

ni → 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 → aval + il → avalil
If end with m then delete m and add tt+ LOC at the end.

mAnam + LOC → mAnam + tt + il → mAnattil
If end with i then add y+ LOC at the end.

rAman + DAT → rAman + u → rAmanu
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 mAR at the end.
\[ Amma + PL \rightarrow amma + mAR \rightarrow ammamAR \]

Rule 3: If end with i then add kal at the end.
\[ mitakki + PL \rightarrow mitakki + kal \rightarrow mitukkikal \]
(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.
\[ sambaNaN + PL \rightarrow sambaNaN \rightarrow sambaNa \]
\[ ila + PL \rightarrow ila + kal \rightarrow ila + kal \]
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.
\[ sambaNaN + PL \rightarrow sambaNaN \rightarrow sam \]

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

<table>
<thead>
<tr>
<th>N- Noun</th>
<th>NP- Pronoun</th>
<th>NCA - Abstract Noun</th>
<th>NCCO – Collective Noun</th>
</tr>
</thead>
<tbody>
<tr>
<td>V- Verb</td>
<td>NCP- Proper Noun</td>
<td>NCC - Common Noun</td>
<td>NPTNN - NanarthakaSarvakam</td>
</tr>
<tr>
<td>NQ- Abstract Noun</td>
<td>NPF - First Person</td>
<td>NPS - Second Person</td>
<td>NPTNC - ChodyaSarvanamam</td>
</tr>
<tr>
<td>NV- Verbal Noun</td>
<td>NPT - Third Person</td>
<td>NC-Concrete Noun</td>
<td>NPTNY - VyakshepakaSarvanamam</td>
</tr>
<tr>
<td>NPTNN NanarthakaSarvakam</td>
<td>NPTNS SarvavachiSarvanamam</td>
<td>VPVM- Munvinayechem</td>
<td>NPTNR NirdhishtavachiSarvanamam</td>
</tr>
<tr>
<td>NPTNM AmsavachiSarvanamam</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
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


