Volume 3, Issue 11, November 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

A Novel Approach for Recognizing Morphological words

G.Pratibha
Asst.Prof(CSE),
Matrusri Engineering College Hyderabad
India

Dr.Nagaratna Hegde
Professor(CSE)
Vasavi College of Engg
Hyderabad
India

A. Haritha
Asst.Prof(CSE)
St Mary's Engg College
Hyderabad
India

M. Sunitha
Asst.Prof(CSE)
Sri Sharada Institute of Sci.&Tech
Hyderabad ,
India

Abstract-Parsing is a well-known process for syntactical analysis of language constructs. Parsers play a vital role in recognizing well defined or classified constructs of a given language. Among the different top down parsing techniques predictive parser is most suitable for recognizing natural language constructs. Indian languages are a real challenge to natural language processing due to their agglutinative nature. Telugu is an Indian language which is morphologically rich and therefore a challenge for morphological analysis. The current paper discusses about the use of predictive parsing technique in Natural Language Processing (NLP) for the recognition of language constructs in telugu.

Keywords: - parsing, predictive parsing, morphology, NLP.

I. INTRODUCTION

Natural Language Processing is an interdisciplinary research area that aims to provide human-computer interaction by processing natural languages.Telugu is most prominent Indian language which contains fifty six alphabets. It is a morphological inflectional Language, Which generates sentences with respect to the context. Telugu is most popular natural language which is highly context sensitive. Words in this language are inflected based on context. The inflected words in sentence can be subject or object or verb. In this paper verb inflections are discussed. Some verbs are similarly inflected due to their verbal behaviour. Such verbs are classified and recognized with the help of a parser. Predictive parser is most suitable to check the syntax and the verbs which belong to the same class. Recognizing different words of Telugu .Which behaves same and belonging to the same class.

II. PARSER

The field of natural language processing has entered its sixth decade. Computers equipped with effective natural language models and process could access all human knowledge of Telugu language in linguistic form. Parser is a syntax-analyser which is used check the Syntax of a program and it recognizes a sentence by consulting a context-Free grammar. Parser can be generated in two ways as top-down parser and bottom-up parser. Parser plays a vital role in generating yield from the context-free grammar. Predictive parser is non-backtracking and non-recursive in the nature. Hence, predictive is the most suitable parser to identify the syntax of a language in top down manner.Parser play a vital role in a compiler to check the syntax of the program. In this paper, how parser is useful to recognize the telugu language verbs which are having same tags when they are placed in a sentence is discussed.Inorder to recognize them a context-free grammar for such verbs is required required.

III. CLASSIFICATION

Classification is a technique to group the things into a class which are similar in nature. Telugu has many verbs which will be with different suffixes or prefixes with respect to context. Here in this paper, Telugu verbs which are with suffix "yu" are classified into a same class due to their same inflection such as "stAdu" when gender of subject is masculine and "stundi" otherwise. Here in this paper, only the present tense is considered to write the context-free grammar. Some telugu verbs which will be inflecting same is given below as cheyu, rayu, koyu, teeyu, veyu, etc.In this example,Only the gender for male is shown. Some example verbs have shown in Fig.1 as cheyu, rayu, koyu, veyu,etc.

Che+yu → che + stAdu
Ra +yu → Ra +stAdu
Ko+yu →ko +stAdu
Tee +yu → Tee+stAdu
Ve+yu →ve+stAdu

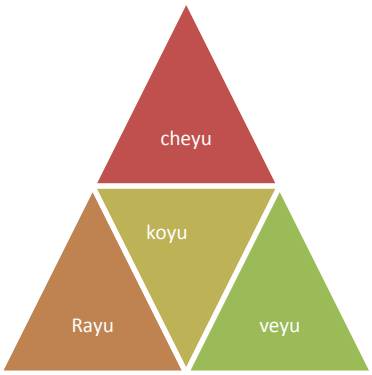

Fig.1 Class of verbs

IV. CONTEXT-FREE GRAMMAR

Context-Free grammar is type-2 grammar in Chomsky classification, which is denoted by G and defined as G= (V, T, P, S), where V, T, P, S are non-terminals, terminals, productions and start-symbol respectively. Context-Free grammar for classifying the verbs is as follows.

Context –Free grammar

P:

S → Ayu

A → VA/CA/€

V →a/e/i/o/u

C → b/c/d/f/g/h/j/k/l/m/n/p/q/r/s/t/v/w/x/y/z

The grammar can derive the verb which ends with yu. It is with total 30 productions which is defined as G=({S,A,V,C},{a,b…z},P,S)

V. PREDICTIVE PARSER

Predictive is a top-down non-recursive and non-backtracking parser which is most efficient in recognizing sentence of grammar. In this paper predictive is used in recognizing similar class of verbs which are similarly inflected. Predictive parser recognizes sentence W as follows.

Algorithm:

Input: A String (W), Predictive parsing table M[X, a]

Output: accept / error action.

Method: If X is the symbol on top of the stack and a current input symbol then

1) If X=a=$ then halt the process and announce it as successful parsing.
2) If X=a≠$ then pop one symbol from top of the stack and advance the pointer to the next symbol in the buffer.
3) If X is a non-terminal and a is current input symbol then check the entry M[X,a] parsing table, if it has X→uvw then replace X by uvw in reverse order such that u should come on top.
4) Other than 1, 2, 3 are called error.

VI. EXPERIMENTAL WORK

Predictive parser recognizes sentence consulting predictive parsing table for the corresponding context-free grammar.pasing table can be constructed as follows.

Algorithm:

Input: A context-Free grammar

Output: A Predictive parsing table.

Method: Create a Two-Dimensional array with number of rows is equal to number of non-terminals and number of columns is equals to number of terminals plus one called M[X, α] as follows.

1) For each production A→α(α≠€),add A→α to M[A,a] where a is First(A)

2)For each production A→€ add A→€ to M[A,b],where b is the symbol generated from Follow(A).

From the above algorithm predictive table can be constructed by calculating FIRST or FOLLOW of a given non-terminal. For instance, productions of the grammar named as 1,2…30 e.t.c.,FIRST(S)={[a-z],$}hence each entry of M[S,α] is filled with 1

TABLE 1

PREDICTIVE PARSING TABLE FOR TELUGU VERBS WITCH ARE WITH SUFFIX yu

M[X,α]	a	b	c	d	e	f	g	h	i	j	k	l	M	n	o	p	q	r	s	t	u	v	w	x	y	z	$
S	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
A	2	3	3	3	2	3	3	3	2	3	3	3	3	3	2	3	3	3	3	3	2	3	3	3	3,4	3	3
V	5				6				7						8						9						
C		10	11	12		13	14	15		16	17	18	19	20		21	22	23	24	25		26	27	28	29	30	

The above table shows the first and follow values of the productions as required by the grammar. This parsing table is seen by the parser when a particular nonterminal on top of the stack and a terminal as the current input symbol. It takes the production in the corresponding entry and is used in performing push operation. This process will be repeated until the stack and buffer is with just $.

VII. RESULTS :

TABLE 2 PREDICTIVE PARSING FOR SENTENCE "cheyu" GENERATION

STACK	INPUT BUFFER	ACTION
$S	**cheyu$**	**Shift S→Ayu**
$uyA	**cheyu$**	**Shift A→CA**
$uyAC	**cheyu$**	**Shift C→c**
$uyAc	**cheyu$**	**pop**
$uyAC	**heyu$**	**Shift C→h**
$uyAh	**heyu$**	**pop**
$uyA	**eyu$**	Push A→VA
$uyAV	**eyu$**	**Push V→e**
$uyAe	**eyu$**	**pop**
$uyA	**yu$**	**Push A→€**
$uy	**yu$**	**pop**
$u	**u$**	**pop**
$	**$**	**accept**

***Parse Tree*:** If the String is accepted by the parser then it generates a tree corresponds to it. Here is the parse tree for the string "cheyu" The string cheyu is from the above specified class. Hence, it is recognized by the parser and generates a parse tree as the result. Parse tree construction starts from start symbol of grammar S and it is extended till it reaches to the required string.

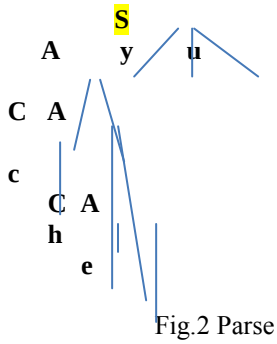

Fig.2 Parse Tree for recognizing sentence "cheyu"

VIII. CONCLUSIONS

While NLP is a relatively recent area of research and application, as compared to other information technology approaches, there has been sufficient success to date. Though it has more success rate, still much more work has to be done yet. Recognizing capability of predictive parser for natural language is discussed in this paper. Classification and recognition of one class is shown here.same is applicable to the remaining verbs also. Ambiguities in this process are eliminated by phrase-level recovery or panic-mode recovery method. In future this method can be used to solve many issues in natural language generati**on.**

REFERENCES

[1] Aho, Alfred &Sethi,Ravi&Ullman,Jeffrey."Compilers: Principles, Techniques, and Tools"
[2] Appel, Andrew "Modern Compiler Implementation in C/JAVA/ML"
[3] Weinberg,G.M. "The Psychology of Computer Programming: "Silver Anniversary Edition.
[4]. Hunter R."The Design and Construction of Compilers" several chapters on theory of syntax analysis, plus discussion of parsing difficulties caused by features of various source languages.
[5]. Alisia Kongthon,Chatchawal Sangkeettrakaran,Sarawoot Kong young and Choochart Haruecheyasak,published by ACM 2009"online Helpdesk system on conversational system"
[6]. Youcong Duan,Christophe cruz "Formalizing Semantics of Natural Language through Conceptualization from Existence"
[7] StevenP.Abney "A Computational Model of Human Parsing"
[8] Steven p.Abney "Part-of –Speech tagging and partial parsing"
[9] Alexander Gelbuch-"Natural Language Processing and its Applications" Research in Computing Science is published by Center for Computing Research of IPN.Volume 46, March, 2010, p.p.no 311-335.
[10] Xiaoyong Liu- "Natural Language Processing", School of Information Studies at Syracuse University, Volume 2,p.p.no1-14

