

Chandhassu Recognizer for Telugu Poems

I. Reddy Sekhar Reddy

*Asst. Prof. in Dept. of CSE
Sree vidyanikethan engineering college
Tirupathi*

M.Humera Khanam

*Asso.Prof. in Dept. of CSE
S V U college of Engineering
Tirupathi*

Abstract : Now a days Natural language processing is one of the most emerging technology in computer science. Challenging topics in this field is processing the Indian languages. The main objective here is to reduce human effort to learn other languages like artificial languages to operate a machines and make a system to understand natural languages. Chandhasu is a basic information to know whether a given poem is syntactically correct or not. Proposed system is help full to check whether a given poem is syntactically correct or not. This system can be developed based on predefined rules for each type of Chandhasu. In Sanskrit Chandhasu types are called vruthas, some vruthas are adapted to write telugu poetry, Vruthas based on number of letters. In Telugu Chandhasu types are called jaathulu and upajaathulu. These are dependent on number of ganalu. And also it checks whether poem follows yathi and prasa properties. Poetry can be useful to represent large amount of information in a small and structured format such a way that it can be easy to remember.

Keywords:Chandhassu, Laguvu, Guruvu, Ganalu, Yathi, Prasa, Labels of Ganalu, Vruthalu, Jathulu, Upajathulu, tokenization, Poems, Unicode, Rule based approach.

INTRODUCTION:

The main objective of this paper is to recognize type of Chandhassu, yathi and prasa letters used in a Telugu poems based on Unicode representation. To accomplish this tasks we need to make a system such that it need to recognize different types of letters present

in Telugu language, system need to understand different properties of Chandhassu types. To achieve this her we are using rule based approach it is one of the NLP technique in syntax analysis. Here rules are defined by linguistic experts.

Natural language processing (NLP) is a subfield of artificial intelligence and linguistics. It studies the problems of automated understanding and generation of natural human languages. Natural language understanding systems convert samples of human language into more formal representations that are easier for computer programs to manipulate and natural language generation systems convert information from computer databases into normal-sounding human language. Language processing applications will use knowledge about language. In theory, natural-language processing is a very attractive method of human-computer interaction .Modern NLP algorithms are grounded in machine learning, especially in rule based and statistical machine learning.

CHANDHASSU:

Chandhassu is a basic information to know whether a given poem is syntactically correct or not. Science of Chandhassu tells about characteristics of poems. It will came from a word called as chadhisamvarani. Characteristics of poems is called chandhassu do to arranging an information in limited

number of letters in particular order and that will very delight to lesion.

There are 26 types of Chandhassu in Sanskrit. Each type of Chandhassu is recognized by the number of akxaras present in each line of the poem. As an akxara can be either a 'laghuvu' or a 'guruvu'. Some corers of vruthas are possible with this 26 types. Among those mostly below two hundred vruthas are regularly used in writing a slokas, among this some vruthas are used in writing a telugu poetry. All this vruthas are dependent on "య మా తా రా జ భా న స ల గం " ganalu. In samavruthas number of lines four, number of letters in each line should be same, second letter in each line call it as a prasa.

యగణము	IUU
మగణము	UUU
తగణము	UUI
రగణము	UIU
జగణము	IUI
భగణము	UUI
నగణము	III
సగణము	IUU
లగం(వగణము)	IU
గగ	UU

In telugu Chandhassu types are called jathulu and upajathulu. This are dependent on indhraganalu and suryaganalu those are

ఇంద్రగణాలు	
నల	III
నగ	IIIU
భగణము	UII
రగణము	UIU
తగణము	UUI
సల	IIUI
సూర్యగణాలు	
నగణము	III
హగణము	UI

SYSTEM ARCHITECTURE:

Chandhassu Recognizer takes input as a telugu poem, check whether a given poem follow any type of Chandhassu rules or not. If it follow Chandhassu rules then it produce which type of Chandhassu it followed, letter used for prasa if it present, letters used in yathimaithri places. If it was not followed any type of Chandhassu then it produce output as syntax error in a given poem. Architecture for RBSA shown in below figure

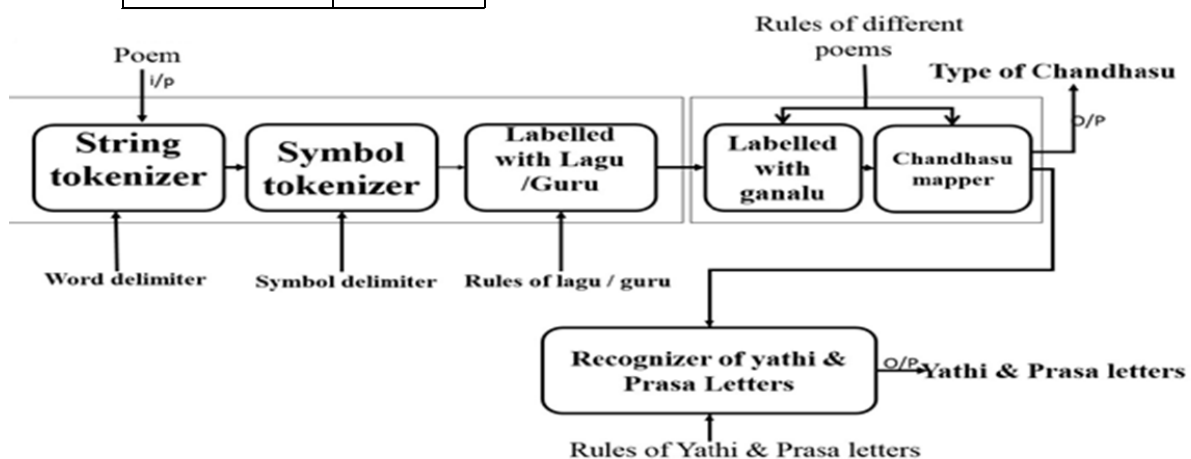


Fig : Architecture of Chandhassu Recognizer(C.R)

Chandhassu Recognizer (C.R) take input as a poem then send it to deciding of lagu or guru stage. In this stage C.R count number of lines, each line divided into words, each word divided into letters then each letter labelled with either lagu or guru. That labelled data send to labelled with ganalu. Hear labelled data is going to be divided into ganalu, each ganam is labelled with appropriate letter from “ya ma tha ra jab ha na sa la gam”. This data is send to Chandhassu mapper it is going to be identify type of Chandhassu and return to user and also send it to recognizer of yathi and prasa letters. It takes poem and poem type as a input and finally it produce prasa letter, yathi mithri placed between the letters.

SYNTAX ANALYZER FOR TELUGU POEMS:

C.R takes input as a Telugu poem, check whether a given poem fallow any type of Chandhassu rules or not. If it fallow then it produce which type of Chandhassu it fallowed, letter used in prasa and yathimaitri places. If it was not fallowed any type of Chandhassu then it produce output as syntax error in a given poem.

Modules:

Following 5 modules are involves in Syntax analyzer for Telugu poems

1. User interface
2. Deciding of Lagu/Guru
3. Divided and Labelled with Ganalu
4. Chandhassu mapper
5. Recognizer of Yathi and Prasa Letters

➤ **User interface**

In this module we create a user page using GUI, which will be the media to interact user with the C.R and through which client can able

to give request to the C.R and C.R can send the response to the user, through this module we can establish the communication between client and C.R. In this page user able to know about the overview of the whole application and having some link to get better knowledge about the whole application.

➤ **Deciding of Lagu/Guru:**

In this module it takes input as a poem and output would be each letter in a poem tagged with either lagu or guru. For this module input space is {all the telugu letters} output space is { U,I }

➤ **String tokenizer :**

String tokenizer takes input as a given poem. Poem divided into words based on word delimiter, Hear’-’.

➤ **Symbol tokenizer :**

Symbol tokenizer takes input words from String tokenizer, each word divided into letters.

➤ **Labelling with Lagu /Guru :**

Labelling with Lagu/Guru takes input as letters, word boundaries from symbol tokenizer and different types of rules regarding lagu/Guru from proposed ‘algorithm 1’,this algorithm works based on Unicode’s, finally it gives output as each letter in given poem labelled with either Laguvu or Guruvu.

Initially given input poem divided into word and letters. Hear each letter need to check all properties defined in algorithm 3.1. To recognize dirgakxaralu, dithva, samyukthaxalu it is going to be use Unicode representations of

respective telugu letters. To recognize polluhallu axaralu initially it converts telugu letter to Unicode then it checks ending Unicode is equal to pollu Unicode.

Algorithm 3.1: Labelling with Lagu/Guru

//Input : poem

//Output: Each letter labelled with Lagu/Guru

Step1: Start

Step2: Divided given poem into words

Step3: Divide each word into letters

//Recognition of

Dirga, purnabindhu, visrgaxaralu, Dithva, samyukthaxaralu and Poluhalluaxaralu based on Unicode Representations listed in 'table 1'

Step4: Letter 'i' is in Dirga, purnabindhu, visrgaxaralu then label with 'U'

Step5: Letter 'i' is Dithva or samyukthaxaramu then previous letter (i-1)

label with 'U' if 'i' is not starting of a word

Step6: Poluhallu occur then label with 'U'

Step7: Other wise label with 'I'

Step8: Stop

➤ **Divided and Labelled with Ganalu:**

Divided and Labelled with Ganalu take input form Deciding of Lagu/Guru module. Main functionality of this module is divide Lagu/Guru into ganas based on Chandhassu types i.e vruthas, jathulu and upajathulu. Each ganam should name based on laguvu or guruvulu present in that ganam.

Chandhassu mapper:

Based on labels generated by Labelling with Lagu/Guru, Chandhassu mapper mapped to appropriate chandhassu this tasks accomplish by following algorithms.

Algorithm 3.2: To Recognize Vruthas(26 types)

//input : Sequence of lagugurvulu labelled by

Algorithm 3.1.

//output : Type of Chandhassu.

Step1: Start

Step2: Check number of lines in the poem is equals to 4 and

number of letters(N) in every line are equal then

Step3: compute $R=N\%3$, $X=N-R$

Step4: Divide 'X' labels into ganas such that each ganam having three labels

Step5: Give name to each ganam based on

"యమాతారాజభానసలగం"

Step6: if $R=1$ then label of $X+1 = 'U'$ then గ otherwise if

$R=2$,

label of $x+1='I'$ & $x+2='U'$ then వ or label of $x+1='U'$ & $x+2='U'$ then గగ

Step7: Repet Step4 to step6 for remaining three lines

step8: Check names of ganas in each line in the poem is equals to "భరనభరవ" and n.of letters in each line equals to 20 then

return "ఉత్పలమాల"

or equals to "నజబజజజర" and n.of letters in

each line equals to 21 then

return "చంపకమాల"

or equals to "సభరనమయవ" and n.of letters in

each line equals to 20 then

return "మత్తేభము"

or equals to "మసజసతతగ" and n.of letters in each line

equals to 19 then

return "శార్దూలము"

or equals to "నజజజజజజవ" and n.of letters in each line

equals to 23 then

return "కవరాజవిరాజతము"

or equals to "నభరసజజగ" and n.of letters in each line

equals to 19 then

return "తరలము"

or equals to "తభజజగ" and n.of letters in each line equals to 14 then
return "వసంతతిలకము"

or equals to "రసజజభర" and n.of letters in each line equals to 18 then
return "మత్తకోకిల"

or equals to "తతజగ" and n.of letters in each line equals to 11 then
return "ఇంద్రవజ్ర"

or equals to "ససస" and n.of letters in each line equals to 12 then
return "త్రోటకము"

or equals to "భభభగ" and n.of letters in each line equals to 11 then
return "దోధకము"

or equals to "సజసజ" and n.of letters in each line equals to 13 then
return "మంజుభాషిణి"

or equals to "నజజయ" and n.of letters in each line equals to 12 then
return "త్రోధకము"

or equals to "యయయ" and n.of letters in each line equals to 12 then
return "బుజంగప్రయాతము"

or equals to "భభభభభభభ" and n.of letters in each line equals to 22 then
return "మానినీ"

or equals to "సజసజసజస" and n.of letters in each line equals to 21 then
return "మణిమాల"

or equals to "రనభగ" and n.of letters in each line equals to 11 then
return "స్వాగత"

or equals to "భజసనగ" and n.of letters in each line equals to 14 then
return "వనమయూర"

or equals to "నజభజజవ" and n.of letters in each line equals to 17 then
return "నర్కట"

or equals to "నననసభభభ" and n.of letters in each line equals to 25 then
return "భంధుర"

or equals to "నభభర" and n.of letters in each line equals to 12 then
return "ద్రుతివిలంబిత"

or equals to "మసజ" and n.of letters in each line equals to 10 then
return "శుద్ధవిరాటి"

or equals to "నసనసనస" and n.of letters in each line equals to 24 then
return "లయవిభాతి"

or equals to "జతజగ" and n.of letters in each line equals to 11 then
return "ఉపేంద్రవజ్ర"

or equals to "రరర" and n.of letters in each line equals to 12 then
return "స్వర్ణిణి"

or equals to "మనయగ" and n.of letters in each line equals to 10 then
return "ప్రణవ"

Step9:step

To recognize vruthas initially it is going to be check basic property of vruthas that is number of letters in each line. If number of letters in each line is equal in a given poem then it decides that a given poem belongs to vruthas, otherwise it was going to be check for other Chandhassu types. To recognize type of Chandhassu for a given input poem after decided as it belongs to vruthas it is going to be divide input labels generated by algorithm 3.1 into ganas such that each ganamu contains three labels. To achieve this number of letters in a line divided with 3 reminder is removed from total number of letters and divide into ganas. Remaining labels can consider it as a one ganamu. To recognize Chandhassu types initially C.R is going to check number of letters. Number of letters is equals to appropriate Chandhassu type letters then it is going to be check for labels of ganalu. Based on labels it is going to be decide type of Chandhassu.

జాతులు మరియు ఉపజాతులు

కంఠము:

Algorithm 3.3: To recognize kandhamu:

- Step1:Start
- Step2:Check number of lines in poem is equals to 4 then
- Step3:Check number of mathras in each line
 - If 'U' occur take it as two mathras
 - If 'I' occur take it as one mathra.
 - If number of mathras in line 1&3 is equals to 12 , 2&4 is equals to 20 then
- Step4:Check first letter label of each line is equal then
- Step5:chek last letter label of 2 and 4 th line is euql to 'U' then
- Step6:Divide into ganalu such that each ganam haveing 4 mathras
- Step5:Label ganalu based on "గగ,బ,జ,స,నల"
- Step6:consider 1&2 into one line , 3&4 into one line
- Step7:Check labels of 1,3,5,7 ganalu not equal to 'జ'
- then
- Step8:check label of 6 ganam is equal to 'నల or జ ' then
- Step9: Return కందము
- Step10:stop

To recognize kandhamu initially RBSA is going to check number of mathras in each line in a given poem. Number of mathras calculated based on lagu guruvulu came from Algorithm 3.1. If 'U' occur count it as a two mathras. If 'I' occurs count it as a one mathra. In kandhamu 1st and 3rd line contains 12 mathras, 2nd and 4th lines contains 20 mathras. Initially it is going to check number of matharas if it satisfies above conditions then only it is going to be check for remaining properties. Kandhamu form based on chathurmathraganalu, those are gaga, bha, ja, sa, nala. First and third line countains three ganalu. Second and fourth lines contains five ganalu. Then it is going to be check first letter label in each line is same are not. After this it is going to be check for second and fourth line last letters labels should be 'U'. Next it is going to be

combine first and second lines into one line. Third and fourth lines into another line. Now it is going to be check for odd ganas labels should not be 'ja' and finally it is going to check for sixth ganam should be 'nala or ja' then RBSA is going to tell 'Kandhamu' is a type of Chandhassu for a given input poem.

ద్విపద:

Algorithm 3.4 : To recognize Dvipadha :

- Step1:Start
- Step2:Check number of lines in poem is equals to 2 then
- Step3:Inshialize ganalu=0
- Step4:If laghu occur then divide 4 labels into one ganame other wise three labels into one ganamu
 - ganalu+1
- Step5:Repet Step4 untill ganalu<3
- Step6:Label ganalu based on "Indhraganalu"
- Step7:Remining labels equals to 'III' then 'స ' or equals to 'UI' then 'హ'
- Step8:Repet step3 to step7 for second line
- Step9:If it satisfied then return "ద్విపదము"
- Step10:Stop

To recognize dvipada type poems initially C.R is going to be check for number of lines in a given poem. If number of lines in a given poem is equals to two then it is going to check for dvipada properties in a given input poem. Dvipada contains three indhraganalu and one suryaganamu in each line. Indhra and surya ganalu represented in

Table 2.4. To divide labels forwarded from algorithm 3.1 into ganas it fallowed this processor. If ganamu starting with 'U' means it is going to be divide three labels into one ganamu or else if ganamu starting with 'I' then it is going to be divide four labels into one ganamu. To give names to ganas it refers indhraganalu from 'Table 2.4'. After completion of first ganam it is

going to check suryaganamu. To Recognize this it is going to check current label is 'U' then two labels, otherwise if current label is 'I' then three labels consider it as a one ganamu. To label ganas with appropriate names it refers suryaganalu from 'Table 2.4'. Finally if a given poem satisfies then it is going to be tell given poem is 'dvipada' Chandhassu type.

ಅಟವಲದಿ:

Algorithm 3.5 : To recognize Ataveladhi :

- Step1:Start
- Step2:Check number of lines in poem is equals to 4 then
- Step3:Inshialize ganalu=0
- Step4:If laghu occur then divide 3 labels into one ganam other wise two labels into one ganamu ganalu+1
- Step5:Repet Step4 until ganalu<3 for 1&3 lines, ganalu<5 for 2&4 lines
- Step6:Label ganalu based on"Suryaganalu"
- Step7:Skip step8 to 11 for 2&4 lines
- Step8:Inshialize ganalu=0
- Step9:If laghu occur then divide 4 labels into one ganame other wise three labels into one ganamu ganalu+1
- Step10:Repet Step4 until ganalu<2
- Step11:Label ganalu based on "Indhrganalu"
- Step12:If it satisfy then return "ಅಟವಲದಿ"
- Step11:Stop

To recognize Ataveladhi initially C.R is going to check number of mathras in second and fourth line in a given poem. Number of mathras calculated based on lagu guruvulu came from Algorithm 3.1. If 'U' occur count it as a two mathras. If 'I' occurs count it as a one mathra. In Ataveladhi 2nd and 4th line contains 15 mathras, Ataveladhi contains three suryaganamulu and two indhrganamulu in first and third line, five

suryaganamulu in second and fourth line. Recognition of suryaganamulu based on number of mathras. If number of mathras is equals to three then make it as a one ganamu up to three ganas in first and third line, up to five ganas in second and fourth line. After that If ganamu starting with 'U' means it is going to be divide three labels into one ganamu or else if ganamu starting with 'I' then it is going to be divide four labels into one ganamu in first and third line only. To give names to ganas it refers indhrganalu from 'Table 2.4'. Finally if a given poem satisfies above conditions then it is going to be tell given poem is 'ataveladhi' Chandhassu type.

ತೆಟಗಿತಿ:

Algorithm 3.6 : To recognize Thetageethi :

- Step1:Start
- Step2:Check number of lines in poem is equals to 4 then
- Step3:If laghu occur then divide 3 labels into one ganam other wise two labels into one ganamu
- Step4:Label ganalu based on"Suryaganalu"
- Step5:If laghu occur then divide 4 labels into one ganame other wise three labels into one ganamu
- Step6:Label ganalu based on"Indhrganalu"
- Step7:Repet Step5 and 6 once againe
- Step8:Repet Step3 and 4 two times
- Step9:Repet Step3 to 8 for all lines in a poem
- Step8:If it satisfy then return "ತೆಟಗಿತಿ"
- Step9:Stop

Thetageethi contains one suryaganamu, two indhrganamulu and then two suryaganamulu in each line. To recognition of suryaganamulu based on number of mathras. If number of mathras is equals to three then make it as a one ganamu . After that If ganamu starting with 'U' means it is going to be divide three labels into one ganamu or else if ganamu starting with 'I' then it is

తరువోజు:

Algorithm 3.9 : To recognize Tharuvoja :

Step1:Start

Step2:Check number of lines in poem is equals to 4 then

Step3:Inshialize ganalu=0

Step4:If laghu occur then divide 4 labels into one

ganame other wise three labels

into one ganamu

ganalu+1

Step5:Repet Step4 unlitl ganalu<3

Step6:Label ganalu based on"Indhraganalu"

Step7:Remining labels euals to'III' then 'స' or equals to

'UI' then 'ప'

Step8:Repet Step3 to Step7 once again

Step9:Repet step3 to step8 for remining lines

Step10:If it satisfie then return "తరువోజు"

Step11:Stop

Tharuvoja contains three indhraganalu, one suryaganamu, three indhraganalu and one suryaganamu in each line. Indhra and surya ganalu represented in Table 2.4. To divide labels forwarded from algorithm 3.1 into ganas it fallowed this processor. If ganamu starting with 'U' means it is going to be divide three labels into one ganamu or else if ganamu starting with 'I' then it is going to be divide four labels into one ganamu up to three ganas. To give names to ganas it refers indhraganalu from 'Table 2.4'. After completion of first three ganas it is going to check suryaganamu. To Recognize this it is going to check current label is 'U' then two labels, otherwise if current label is 'I' then three labels consider it as a one ganamu. To label gans with appropriate names it refers suryaganalu from 'Table 2.4', after this it is going to make next three ganas are indhra and final one is suryaganamu in all four lines. Finally if a given poem satisfies above

conditions then it is going to be tell given poem is 'Tharuvoja' Chandhassu type.

మధ్యాక్కర:

Algorithm 3.10 : To recognize Madhyakkara :

Step1:Start

Step2:Check number of lines in poem is equals to 4 then

Step3:Inshialize ganalu=0

Step4:If laghu occur then divide 4 labels into one

ganame other wise three labels

into one ganamu

ganalu+1

Step5:Repet Step4 unlitl ganalu<2

Step6:Label ganalu based on"Indhraganalu"

Step7:Remining labels euals to'III' then 'స' or equals to

'UI' then 'ప'

Step8:Repet Step3 to Step6 once again

Step9:Repet step3 to step8 for remining lines

Step10:If it satisfie then return "మధ్యాక్కర"

Step11:Stop

Mdhyakkara contains two indhraganalu, one suryaganamu and two indhraganalu in each line. Indhra and surya ganalu represented in Table 2.4. To divide labels forwarded from algorithm 3.1 into ganas it fallowed this processor. If ganamu starting with 'U' means it is going to be divide three labels into one ganamu or else if ganamu starting with 'I' then it is going to be divide four labels into one ganamu up to two ganas. To give names to ganas it refers indhraganalu from 'Table 2.4'. After completion of first two ganas it is going to check suryaganamu. To recognize this it is going to check current label is 'U' then two labels, otherwise if current label is 'I' then three labels consider it as a one ganamu. To label gans with appropriate names it refers suryaganalu from 'Table 2.4', after this it is going to make next two ganas

are indhrganamulu in four lines. Finally if a given poem satisfies above conditions then it is going to be tell given poem is 'Mdhyakkara' Chandhassu type.

Recognition of yati and prasa letters:

- In telugu poems second letter of each line will called as prasa.
- Prasa means Hallu should be same but if you want to change only Achu can be changeable.

Example represented in 'Appendix 1' .

Eg:

మా ,మీ ,మే

Hear hallu is మ but achu changed in each letter ఆ,ఈ,ఏ

APPLICATIONS:

C.R is very use full to who are working on Telugu literature to identify characteristics of a particular poem those are type of Chandhassu used, yathi and prasa letters used in a poem. It is very useful in avadhanalu. In avadhanam poet tell poem on a given topic but we doesn't know whether he fallowed Chandhassu rules or not to identify poem properties RBSA is very useful. Where ever the Chandhassu is used in all that cases to identify those Chandhassu RBSA is usefull.

Main applications of Chandassu is to write a poems, songs, prabhandas (collection of poems), dwipada (poems that follow a typical pattern), Sataka (collection of 100 poems)etc. By using poetry we can able to effectively represent sthaya bhavalu (symptoms). Poetry is helpful to represent science of Chandhasu, vyakarana, alnkara, sahithyamu. Poetry is also used to represent dictionaries , medicine ,math's , testing of vasthu ,pearls ,authentication purpose in data transition by using prasa property.

FUTURE ENHANCEMENTS

- If we make speech processing interface to C.R then we can able to perform E-Avadhanams in Telugu poetry.
- Identify a given poem is nearer to witch type of Chandhassu where you need to modify.
- System need to predict appropriate word in writing a poem that will suitable to specified Chandhassu and also it will give same meaning as mention word.

CONCLUSION:

The proposed syntax analyzer for telugu poems is a simple and efficient method to recognize type of 'Chandhassu' for a given input poem and also to recognize 'Yathi' and 'Prasa' letters. The RBSA works based on algorithms proposed for different types of 'Chandhassu', those algorithms are constructed based on rules defined by a linguistic experts.

REFERENCES :

- [1] Thelugu Bhashalo Chandhorethulu By Ravuri Dhorasamisharma
- [2] Rule Based Syntax Analyzer for Telugu Poems by Mr. I. Reddy sekhar Reddy
- [3] Sulaxana Saramu By Limgamagumta Thimmakavi
- [4] Thelugu Vyakharanamu By Raghavendra Publications
- [5] Introduction To Machinelearning By Ethem Alpaydn
- [6] Speech And Language Processing By Daniel Jurafsky And James H.Martin