LANGUAGE IN INDIA

Strength for Today and Bright Hope for Tomorrow

Volume 6 : 8 August 2006

TELUGU PARTS OF SPEECH TAGGING IN WSD

T. Sree Ganesh, M.A., Ph.D.

TELUGU PARTS OF SPEECH TAGGING IN WSD

T. SREEGANESH, M.A., M.Phil.

In modern Linguistics trend 'Corpus Analysis' is a remarkable stream. 'Corpus Analysis' is useful in any area of Linguistics like Phonology, Morphology, Syntax, Socio-linguistics, Machine Translation and Computational Linguistics and so on. It focuses on the description of quantitative patterns of Linguistic elements. Mainly it focuses on the Description of Linguistic Performance in a particular language. Today linguistics believes that what people actually use is the real language. It reflects on the ideological as well as technological meanderers in Linguistics. Ideological changes makes from the path of intuition based rationalistic assumption. The technological change i.e. computers posses and delivers massive storage facilities and impressive processing. In this paper we deal the corpus analysis for parts of speech tagging rules in technological way. For that we observe the definitions to 'corpus-corpora'.

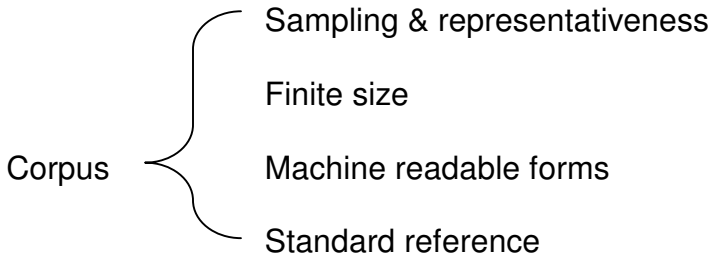
CORPUS-CORPORA

The word corpus is derived from the Latin word, which means 'body'. Corpus is a limited sized body of machine readable texts sampled in order to be maximally representative of the language variety under consideration. It has the quality of 'representativeness.' There are many definitions to corpus given by Linguists.

> Corpus is a collection of naturally occurring texts chosen to characterize a state or variety of language (Sinclair 1991).

> Corpus is collection of running texts which may be spoken, written or intermediate forms and the sample may be of any length (Jan Aarts 1991).

Depending upon these types of definitions we can find that there are mainly four readings to corpus. They are



A collection of huge corpus is considered as corpora. Corpora are the plural form of Corpus. These corpora also have the same four main headings.

Corpus is different from Text.

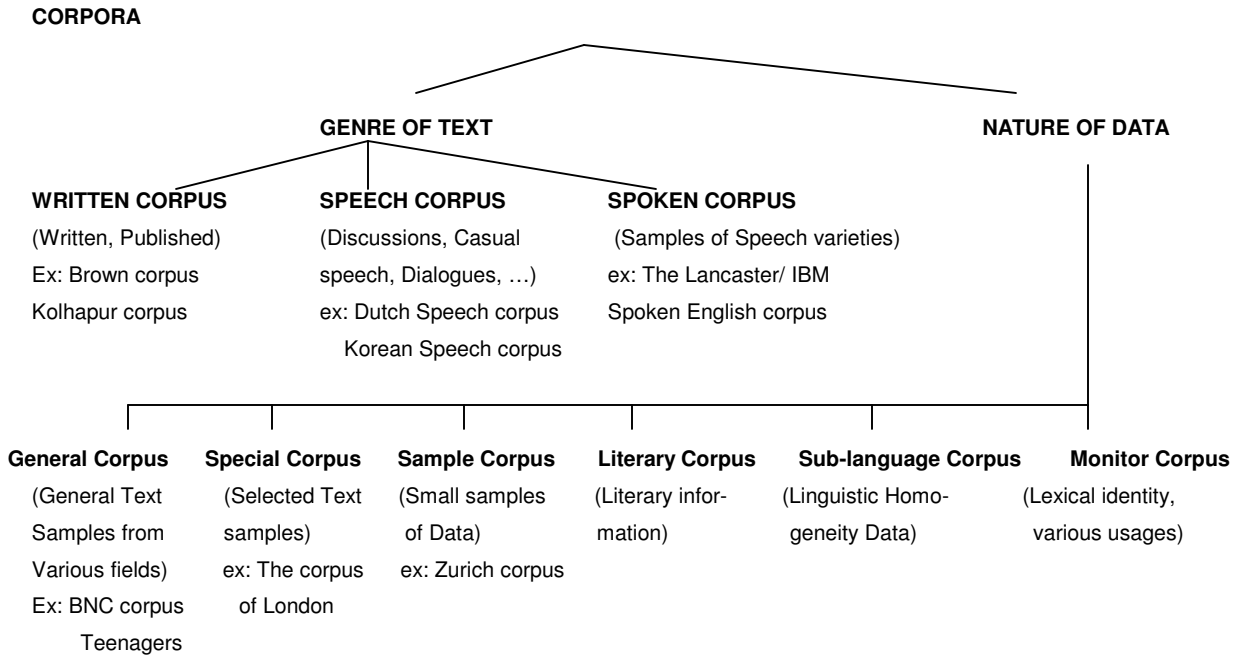
1. Texts are representatives of unified communicative events. Corpus is a fragment.
2. A text is read horizontally from left to right paying attention to the boundaries of units. Corpus may be vertical.
3. Text is unique and individualistic, but Corpus is repeated and social.

But there are limitations in corpus study. They are:

1. Lake of linguistic generativity
2. Technical difficulties
3. Lack of texts from dialogic interactions
4. Lack of information from visual elements.

TYPES OF CORPORA

In Corpora the types include texts as well as the combinations. It is very difficult to design an organized scheme of corpus classification based on the content. With the features, which are discussed below Corpora are classified into the following way by usage.



Till date there is no proper annotated corpus for literary corpus, sub-language and monitor corpus. Especially for Indian languages there are no fully annotated Corpora except Bengali language.

USES OF CORPORA

In modern linguistics 'corpus' study is very essential part to do any research activity. Corpus studies show the total structure of a particular language before our eye. Corpus studies are very useful in speech research, lexical studies, psycholinguistics and NLP and so on. The main advantages of Corpora are:

1. SAMPLING AND QUANTIFICATION

Corpus is a sample of maximum representation of a particular language. Quantification in corpus linguistics is of more meaningful. By sampling and quantification it will help to know about a language.

2. EASE OF ACCESS USING

Once the corpus is ready, as in machine-readable form, it is also easy to access the data. With a program we can quickly retrieve the frequency lists and indices of various words or other structure of language with in it.

3. ENRICHED DATA

English corpora are available with additional linguistics information like POS Tagging, parsing. Data retrieval from the corpus can be easier and more specific than with unannotated data.

4. NATURALISTIC DATA

Corpora should be naturalistic data. Then only it provides the most reliable source of data on language as it is actually used.

These are the main advantages of corpora. In this paper we want to try to build the Parts of Speech Tagging rules for Telugu based on the Corpus, which is available in Telugu.

PARTS OF SPEECH TAGGING

Tagging is nothing but labeling. POS Tagging means we add the Parts of Speech category to the word depending upon the context in a sentence. It is also known as Morpho-syntactic Tagging.

In NLP, at the starting stage to any language application, two tools are necessary. They are: 1.Morphological Analyzer and 2. POS Tagger. These two tools are more important for any Natural Language Processing. Another thing here we have to mention is, POS Tagging is also very essential in Machine Translation to understand the Target Language.

Paanini describes Parts of Speech in Sanskrit by giving the sutra as "*na:ma:Kya:to:pasarganipa:ta:Sca*".

In this he classified the parts of speech into four types. They are

1. *na:maM,*
2. *a:Kya:taM,*
3. *nipa:tam*
4. *upasarga*.

For English language there are eight Parts of Speech like

noun,
pronoun,
verb,. …

Further in POS, we classify the categories into sub-categories in deep analysis of the languages.

TAGS

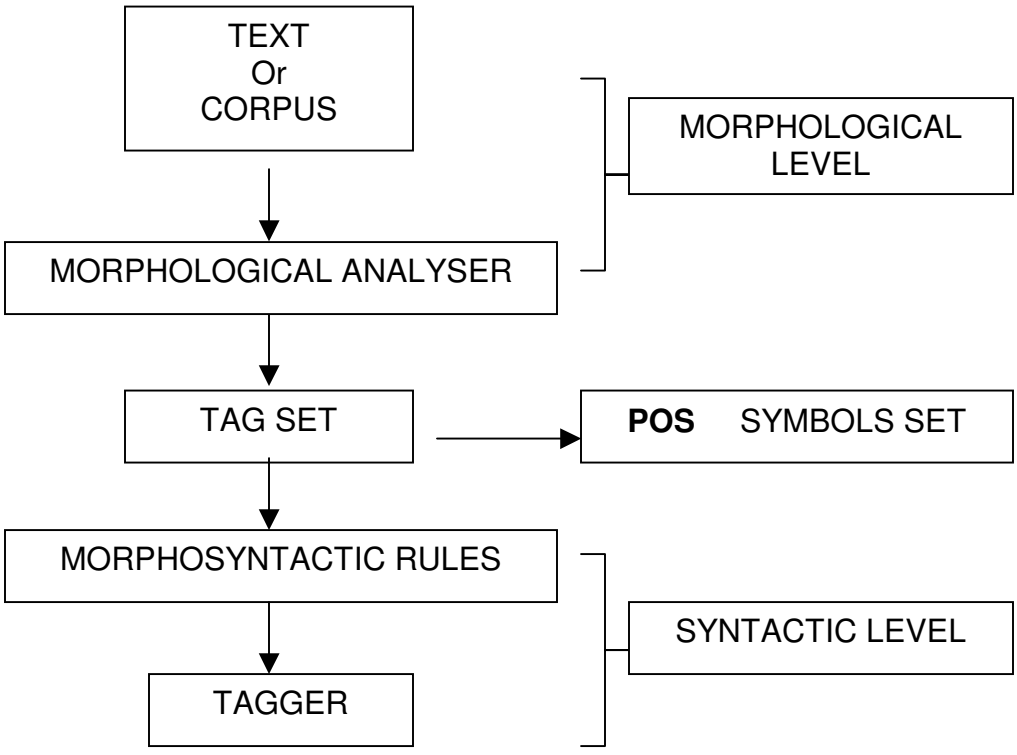
With help of Tags, we develop the "Telugu Tag Set". In Telugu there is mainly five POSes. They are

1. *Noun*
2. *Pronoun*
3. *Verb*
4. *Adjective*
5. *Avyea (ayaya)*.

We make a sub-classification of these main POS further depending upon the context.

RESOURCES FOR POS TAGGING

For POS Tagging it is necessary to use some resources. They are:



These five are the major resources for POS Tagging.

1. TEXT & CORPUS

The Text or Corpus for POS Tagging should be pre-edited and error free. Then only we can use the Text or Corpus for language research. Otherwise there

may be some problems in using the Tagger. So we have to prepare the text for the analysis.

2. MORPHOLOGICAL ANALYZER

Morphological analyzer is a basic tool in any NLP Application. Without morphological analysis we can't go for any computational programming for Indian languages. Some languages particularly Indian languages without morphological analyzer it is very difficult to do any research. By nature many of the languages in India are inflected. For English language the morphological analyzer is easy but not the syntactic analyzer.

For POS Tagging to Telugu, we are using a morphological analyzer, which is developed by G. Umamaheswara Rao from University of Hyderabad. It gives the way to analyze each and every word in the Telugu corpus with the different possibilities.

TAG SET

Tag set is a set, which have different Tags for a particular language. In the first stage manually we will tag different Texts and build a Telugu Tag set. In the present application a Telugu Tag set which have 53 different Tags. These Tags can cover all the POS in Telugu Language. With use of the Symbols we can write the Morpho-syntactic rules. Telugu POS Tag Set is like this.

No.	Description	Code	Example
1.	Noun	NNI	bAludu, naxi
1.1	Noun	NN1	bAludu
1.2	Noun	NN2	bAludini
1.3	Noun	NN3	bAludiwo
1.4	Noun	NN4	bAludikoVraku,
1.5	Noun	NN5	bAludinuMdI
1.6	Noun	NN6	bAludi/bAlura
1.7	Noun	NN7	Imtlo, iMtipEna, iMtimIxa
1.8	Noun	NN8	O, oyI, osI
2.	Compound Proper Noun	NPC	atala/ NNPC bihArI/ NNPC vAjapayI/ NNP SrI/ NNC pI./ NNPC ke./ NNPC misrA/ NNP
3.	Compound Common Noun	NNC	keMxra/NNC praBuwvaM/NN bAlakulu/NNC bAliklu/NN
4.	LocativeNouns		
4.1	LocativeNouns	NL1	moVxata
4.2	LocativeNouns	NL2	moVxatini

4.3	LocativeNouns	NL3	moVxatiwo
4.4	LocativeNouns	NL4	moVxatikosaM
4.5	LocativeNouns	NL5	moVxatinuMdI/ kaMte
4.6	LocativeNouns	NL6	moVxati
4.7	LocativeNouns	NL7	moVxatilo/moVxatiki
4.8	LocativeNouns	NL8	moVxata

5. Postposition

5.1	Postposition	PP1	du, mu, vu, lu
5.2	Postposition	PP2	nu/ni
5.3	Postposition	PP3	wo/cewa
5.4	Postposition	PP4	koVraku/kosaM
5.5	Postposition	PP5	nuMdI/kaMte/valana
5.6	Postposition	PP6	yoVkka
5.7	Postposition	PP7	lo/pE/na
5.8	Postposition	PP8	O, oyI, osI

6. Pronoun

6.1	Pronoun	PR1	nenu, iwanu, awanu, nuvvu
6.2	Pronoun	PR2	vIrini, vAlYlnu, vArini
6.3	Pronoun	PR3	iwaniwo, nAcewa, nIwo
6.4	Pronoun	PR4	iwaniki, awanikoVraku
6.5	Pronoun	PR5	iwanivalana, ImeVkaMte
6.6	Pronoun	PR6	iwaniyoVkka
6.7	Pronoun	PP7	nIyaMxu

7.	Adjective	JJ	nallani, aMxamEna,
8.	Verb Finite	VF	koVttAdu, winnAdu
9.	Verbal Nonfinite	VNF	Vesi, poi,_peVtti
10.	Verbal Adjectival	VJJ	wiMtU, ceVbuwU
11.	Verbal Adverbial	VRB	wini, veVlYli - kwvArWakM
12.	Verbal Nominal	VNN	wAgadaM, winadaM

13. Conjunct

13.1	Conjunct	CA	mariyu
13.2	Conjunct	CY	leka
13.3	Conjunct	CC	ani
13.4	Conjunct	CB	kanI
13.5	Conjunct	EP	gAnI

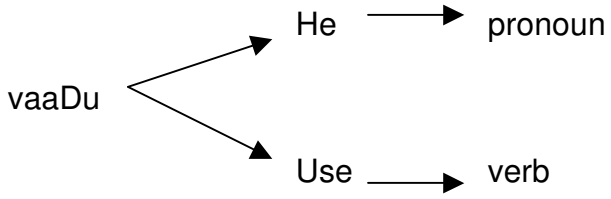
14.	Adverb	RB	meVllagA, woVMxaragA
15.	Particle	RP	LA, varaku, ke, ayiwe, kUdA
16.	Question Words	QW	emiti, eVlA
17.	Quantifier	QF	eVkkuva, koVMceVM
18.	Number Quantifiers	QN	mUdava, mugguru,mUdu
19.	Intensifier	IF	cAlAeVkkuva, iMkAkoVMceVM *
20.	Negative	NG	kAxu, lexu
22.	kriyAmUla nAmaM	NV	snAnaM]nv ceswAdu snAnaM ceswU snAnaM cesi snAnaM cesinaco

MORPHO-SYNTACTIC RULES

Morpho-syntactic rules are necessary to give the proper POS Tag to each word. These rules are formed depending upon the context where the word or groups of words occur in a sentence.

Ex:	In Telugu	vaaDu	vaccaaDu
	Meaning	'He	came'

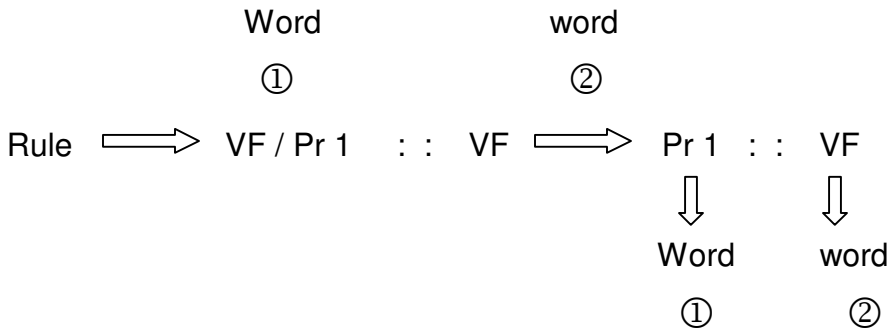
Here 'vaaDu' has two meanings.



The morphological level vaaDu has TWO types of TAGS. So in this stage the sentence can be tagged as shown below:

vaaDu / VF, Pr1 vaccaaDu / VF - / sym

Here the rule for this sentence is given as:



If the word ① has the possibility to have 'VF / Pr 1' Tag after the word ② Have 'VF' Tag, then word ① is 'Pr 1' only. This is our POS Tagging Rule. Sometime these rules may be failed. The context may need to be suited to our rules. In that case we have to develop the thematic rules. This is the logic of our rule formation.

TAGGER

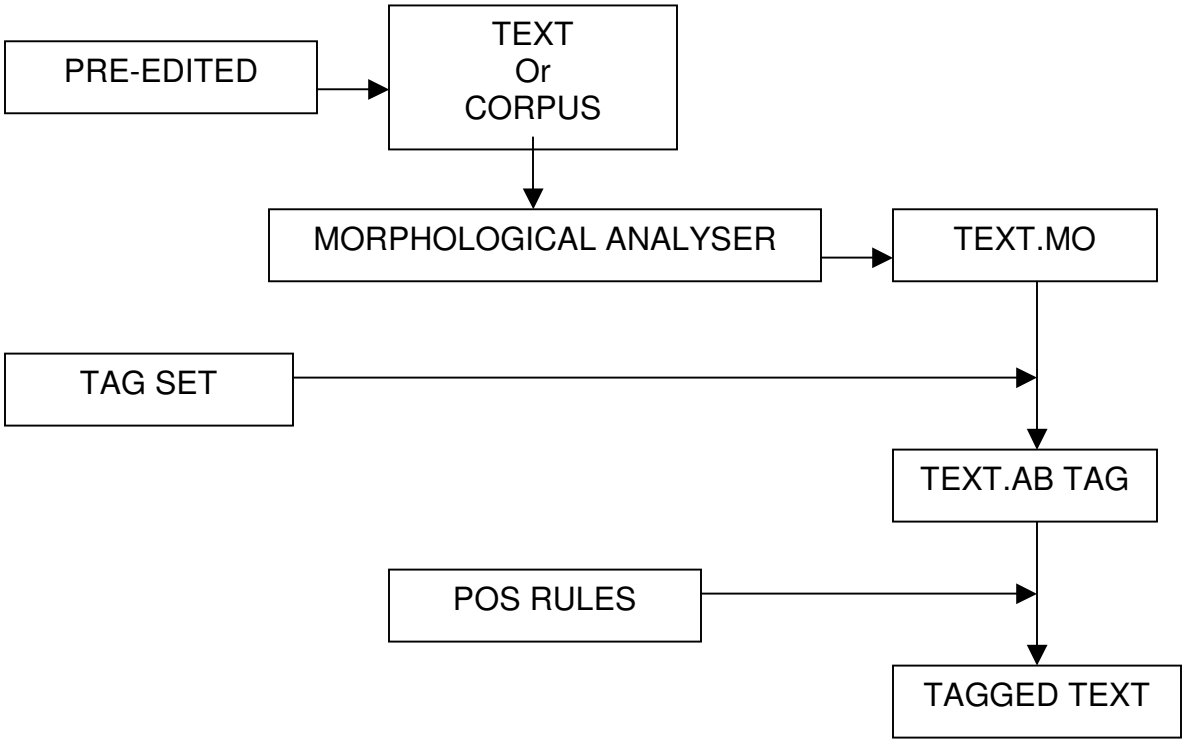
Tagger is a tool, which assigns the POS to each and every word in the CORPUS or TEXT. It works with the help of our programme. POS Tagger is developed on Linear Platform with PERL Language.

In this paper mainly we have concentrated on the Rule Based POS Tagging. There are many ways in Tagging like Statistical Tagging and so on… Rule Based Tagging is somewhat better in these methods. But it is difficult to formulate rules. We have formulated the rules for the TEXT, which we have selected. Basically these rule formations are mainly depend upon the 'Morpho-syntactic' informations. With the use of these rules the POS Tagger add the appropriate POS Tags to each and every word. Now we can observe the Methodology involved in Tagging.

METHODOLOGY

The following diagram will show the procedure involved in POS Tagging for Telugu.

For POS Tagging it is necessary to use some resources. They are:



In the first stage we take a TEXT or CORPUS. The Text we have selected for the testing is a Novel entitled "***haahaa huuhuu***" written by VISHWANATHA SURYANARAYANA. We have not given the whole story for the analysis. We choose one paragraph from the novel for the purpose of explaining the POS in Telugu language. The TEXT, which is selected for this purpose, is given below:

> *oVka rojuna uxayaM laMdanulo treVPAlGar skver vaxxa janaM guMpulu guMpulugA mUgivunnAru. vAru oVka viciwra jaMwuvu vaMka cUswunnAru. A jaMwuvu vArini anni viXAlA AkarRiMciMxi. A jaMwuvu meVdavaraku maniRi. wala mAwraM gurrapu wala.*

Now we send the text to Telugu Morphological Analyzer, which was developed by G.Umamaheswara Rao, CALTS, University of Hyderabad. Then we take the output of Morphological Analyzer as " Text.mo". Then Text.mo is like this:

oVka{lewa n *adj_0* }/
roju{meku n eka *na* }/
uxayaM{puswakaM n eka *0* }/
laMdanu{meku n bahu *0_o* }/ laMdanu{meku n eka *lo* }/
treVPAlGar{kalcar n eka *0* }/ treVPAlGar{kalcar n eka *obl* }/
skver{kalcar n eka *0* }/ skver{kalcar n eka *obl* }/
vaxxa{vaxxa n *adv_0* }/ vaxxa{vaxxa n *adv_yoVkka* }/

janaM{puswakaM n eka *0* }/ jana{SreRTa n *adj_xi_na* }/
guMpu{meku n bahu *0* }/ guMpu{meku n bahu *vu* }/
guMpu{meku n bahu *gA* }/
mUgu{uMdu_deri v *i_uMdu_A* 23_ba }/.
vAru{vAru P bahu *0* }/ vAru{vAru P bahu *vu* }/
oVka{lewa n *adj_0* }/
viciwraM{puswakaM n eka *obl* }/
jaMwuvu{pUvu n eka *0* }/ jaMwuvu{pUvu n eka *obl* }/
vaMka{kaxa n *avy_0* }/ vaMka{kota n eka *0* }/ vaMka{kota n eka *obl* }/
cUdu{cUdu v *wunn* 23_ba }/.

A{AVY Avy }/
jaMwuvu{pUvu n eka *0* }/ jaMwuvu{pUvu n eka *obl* }/
vAru{vAru P bahu *ni* }/
anni{anni P bahu *0* }/ aMwa{aMwa n eka *lAti* }/
viXaM{puswakaM n bahu *0_A* }/
AkarRiMcu{cUpiMcu v *A* 3_non_pu_e }/.

A{AVY Avy }/
jaMwuvu{pUvu n eka *0* }/ jaMwuvu{pUvu n eka *obl* }/
meVda{kota n eka *varaku* }/
maniRi{maniRi n eka *0* }/ maniRi{maniRi n eka *obl* }/.

wala{kota n eka *0* }/ wala{kota n eka *obl* }/ walaM{puswakaM n eka *obl*
}/wa{kota n bahu *obl* }/
mAwraM{AVY Avy }/ mAwraM{pApaM n *avy_0* }/
gurraM{puswakaM n eka *obl* }/
wala{kota n eka *0* }/ wala{kota n eka *obl* }/ walaM{puswakaM n eka *obl* }/
wa{kota n bahu *obl* }/.

Then we add our Telugu Tag set to the Morph output (Text.mo). Then the output will be like this:

oVka{lewa n *adj_0* }/jj
roju{meku n eka *na* }/nn7
uxayaM{puswakaM n eka *0* }/nn1
laMdanu{meku n bahu *0_o* }/nn1
laMdanu{meku n eka *lo* }/nn7
treVPAlGar{kalcar n eka *0* }/nn1
treVPAlGar{kalcar n eka *obl* }/nni
skver{kalcar n eka *0* }/nn1
skver{kalcar n eka *obl* }/nni
vaxxa{vaxxa n *adv_0* }/rb
vaxxa{vaxxa n *adv_yoVkka* }/rb
janaM{puswakaM n eka *0* }/nn1
jana{SreRTa n *adj_xi_na* }/jj

```
guMpu{meku n bahu *0* }/nn1
guMpu{meku n bahu *vu* }/nn1
guMpu{meku n bahu *gA* }/rb
mUgu{uMdu_deri v *i_uMdu_A* 23_ba }/vf
./sym

vAru{vAru P bahu *0* }/pr1
vAru{vAru P bahu *vu* }/pr1
oVka{lewa n *adj_0* }/qn
viciwraM{puswakaM n eka *obl* }/jj
jaMwuvu{pUvu n eka *0* }/nn1
jaMwuvu{pUvu n eka *obl* }/nni
vaMka{kaxa n *avy_0* }/avy
vaMka{kota n eka *0* }/nn1
vaMka{kota n eka *obl* }/nni
cUdu{cUdu v *wunn* 23_ba }/vf
./sym

A{AVY Avy }/jj
jaMwuvu{pUvu n eka *0* }/nn1
jaMwuvu{pUvu n eka *obl* }/nni
vAru{vAru P bahu *ni* }/pr2
anni{anni P bahu *0* }/pr1
aMwa{aMwa n eka *lAti* }/avy
viXaM{puswakaM n bahu *0_A* }/rb
AkarRiMcu{cUpiMcu v *A* 3_non_pu_e }/vf
./sym
A{AVY Avy }/jj
jaMwuvu{pUvu n eka *0* }/nn1
jaMwuvu{pUvu n eka *obl* }/nni
meVda{kota n eka *varaku* }/nl1
maniRi{maniRi n eka *0* }/nn1
maniRi{maniRi n eka *obl* }/nni
./sym

wala{kota n eka *0* }/nn1
wala{kota n eka *obl* }/nni
walaM{puswakaM n eka *obl* }/nni
wa{kota n bahu *obl* }/nni
mAwraM{AVY Avy }/avy
mAwraM{pApaM n *avy_0* }/qf
gurraM{puswakaM n eka *obl* }/nni
wala{kota n eka *0* }/nn1
wala{kota n eka *obl* }/nni
walaM{puswakaM n eka *obl* }/nni
wa{kota n bahu *obl* }/.nni
```

./sym

In this stage we add all the possibilities of different POS Tags to the Original Text with programming. The output file shape is

> OVka/jj rojuna/nn7 uxayaM/nn1 laMdanulo/nn1,nn7 treVPAlGar/nn1,nni skver/nn1,nni vaxxa/rb janaM/nn1 guMpulu/nn1 guMpulugA/nn1,rb mUgivunnAru/vf ./sym vAru/pr1 oVka/qn viciwra/jj jaMwuvu/nn1,nni vaMka/avy,nn1,nni cUswunnAru/vf ./sym A/jj jaMwuvu/nn1,nni vArini/pr2 anni/qn viXAlA/rb AkarRiMciMxi/vf ./sym A/jj jaMwuvu/nn1,nni meVdavaraku/nl1 maniRi/nn1,nni ./sym wala/nn1,nni mAwraM/qf,avy gurrapu/nni wala/nn1,nni ./sym

In this stage with the help of Morpho-syntactic Rules we will disambiguate the ambiguity in POS. For this Novel we have formatted 524 morpho-syntactic rules for POS Tagging disambiguation. But all the rules are not possible in this paper. We are giving some rules, which are used to disambiguate the Text given in this paper. It is sure all the rules may not be possible to disambiguate the POS in any context. We will discuss about this in the problems. Those morpho-syntactic rules are

nn1,nn7 :: nn1,nni => nn7 :: nn1,nni

nn1,nni :: nn1,nni => nni :: nn1,nni

nn1,nni :: rb => nn1 ::rb

nn1,rb :: vf => rb :: nn1

avy,nn1,nni :: vf => avy ::vf

nn1,nni :: pr2 => nn1 :: pr2

nn1,nni :: nl1 => nn1 :; nl1

nn1,nni :: ./sym => nn1 :: ./sym

qf,avy :: nn1,nni => qf :: nn1,nni

With these rules we write a programme to dissolve the POS Tagging ambiguity. The output of the programme is disambiguated POS Tagged Text. That is,

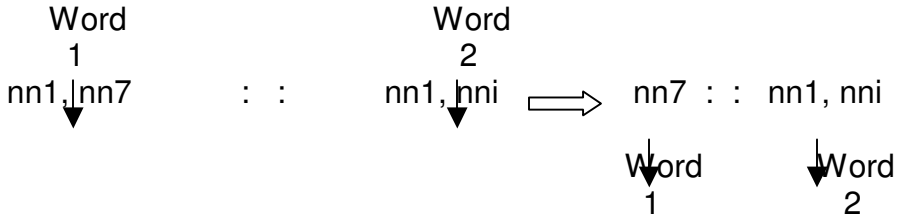
> OVka/jj rojuna/nn7 uxayaM/nn1 laMdanulo/nn7 treVPAlGar/nni skver/nn1 vaxxa/rb janaM/nn1 guMpulu/nn1 guMpulugA/rb mUgivunnAru/vf ./sym vAru/pr1 oVka/qn viciwra/jj

jaMwuvu/nn1,nni vaMka/avy cUswunnAru/vf ./sym A/jj jaMwuvu/nn1 vArini/pr2 anni/qn viXAlA/rb AkarRiMciMxi/vf ./sym A/jj jaMwuvu/nn1 meVdavaraku/nl1 maniRi/nni ./sym wala/nn1mAwraM/qf gurrapu/nni wala/nn1./sym

RULES FORMATION MANNER

In the ambiguity POS Tagged Text, first ambiguity word is

Word 1 ⟶ laMdanulo / nn1, nn7
Next word is
Word 2 ⟶ treVPAlGar / nn1, nni
Our rule is



If the word 1 has nn1, nn7 possible tags and word 2 has the nn1, nni possible tags. Then the word 1 must be " nn7 " only. The word 2 should be remaining as it is. Then the tagger takes word 2 as word 1 compare with following word with our Morpho-syntactic POS Tagging Rules. Then it is also has the proper POS Tag. In this way the TAGGER decide the proper POS Tags to the TEXT.

PROBLEMS

In our practical POS Tagging we face some problems where the POS Tagger does not tag some words. We find out those are in need of ontological and thematical informations. If we provide that information to the system in the form of rules, we hope that the Tagger may be able to solve the problem.

Here our rule is

JJ, Pr6 : : nn1, nni ⟹ JJ : : nn1, nni
With this rule the Tagger [illegible]e that vaaDi is "JJ ". But

Here the Tagger automatically decides that vaaDi is " JJ". This Tagging is contextually wrong. So if we provide the ontological and thematical information then only the Tagger will Tag proper POS Tag.

The information is like this if vaaDi belongs to a noun which have metalized objects. Then those objects have sharpness. That means metalized objects have the sharpness quality. This is the logic related to the ontological and thematical information. Based upon this logic if we write ontological and thematical rules then it is possible to solve these types of problems.

If these types of rules are failed then it is better to apply statistical approach.

CONCLUSION

In NLP POS Tagging is the major task. When the machine understands the TEXT then it is ready to do any NLP applications. For that the machine should understand each and every word with its meaning and POS. This is the main aim of our research. This goes to Morpho-syntactic level. Particularly in MT when the system understand the POS of source language Text then only it will translate into target language without any errors. So POS Tagging plays such as an important role in NLP.

REFERENCES

Tony McEnery and Andrew Wilson (2001) **CORPUS LINGUISTICS - An Introduction,** 2nd Edition, Edinburgh University Press Ltd, Edinburgh.

Niladri Sekhar Dash (2005) **CORPUS LINGUISTICS AND LANGUAGE TECHNOLOGY WITH REFERENCE TO LANGUAGES,** 1ST Edition, Mittal Publications, New Delhi.

Shanmugam, C. (-------), **CORPUS LINGUISTICS,** Annamalai University DDE NLP PG Diploma Course Material, Annamalai Nagar.

Sree Ganesh Thottempudi (2005), **IMGILSHU TELUGU YAMTRAANU VAADA SANDARBHAMGAA TELUGU PADAALA BHASHAABHAGA NIRNAYAM,** M.Phil. Dissertation, Submitted to Department of Telugu, University of Hyderabad, Hyderabad.

T. SREEGANESH
University of Hyderabad
Hyderabad, A.P.
India
mrthottempudi@yahoo.com