


A Modified Markov-Based Maximum-Entropy Model for POS Tagging of Odia Text

Sagarika Pattnaik, Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed), India*

 <https://orcid.org/0000-0003-0664-4520>

Ajit Kumar Nayak, Department of Computer Science and Engineering, Siksha 'O' Anusandhan (Deemed), India

ABSTRACT

POS (parts of speech) tagging, a vital step in diverse natural language processing (NLP) tasks, has not drawn much attention in the case of Odia, a computationally under-developed language. The proposed hybrid method suggests a robust POS tagger for Odia. Observing the rich morphology of the language and unavailability of sufficient annotated text corpus, a combination of machine learning and linguistic rules is adopted in the building of the tagger. The tagger is trained on tagged text corpus from the domain of tourism and is capable of obtaining a perceptible improvement in the result. Also, an appreciable performance is observed for news article texts of varied domains. The performance of the proposed algorithm experimenting on Odia language shows its manifestation in dominating existing methods like rule based, hidden Markov model (HMM), maximum entropy (ME), and conditional random field (CRF).

KEYWORDS

CRF, F Score, HMM, Hybrid, ME, NLP, Rule Based, Tagger

INTRODUCTION

Parts of speech tagging (POS) tagging is an indispensable tool to address various issues of Natural Language Processing (NLP). It is the problem of finding a way to tag every token in a text to a particular part of speech, like noun, verb, adverb, or any other lexical class. It thereby provides information about the token and relationship of these tokens with their adjacent tokens in a sentence (Tiwari and Siddiqui, 2008). Most tokens occurring in a text might have ambiguity associated with them in terms of their grammatical classification. For example the English word “book” can be a noun or a verb depending on the context. Computationally developed languages like English have rich knowledge base (voluminous training corpus, linguistic rules, NLP tools etc.) and solve the ambiguity by various methods like rule based, Hidden Markov Model (HMM), Support Vector Machine (SVM), Maximum Entropy (ME) etc. that are broadly divided under rule based and stochastic approach (Kanakaraddi and Nandyal, 2018). But in context with the languages like Odia that are devoid of sufficient knowledge base for disambiguation, the task becomes challenging for NLP researchers. As POS tagging is the basis of number of NLP tasks like automatic text summarization and information retrieval by determining significant words, word sense disambiguation by deciding the context, text to speech conversion,

DOI: 10.4018/IJDSST.286690

*Corresponding Author

This article published as an Open Access Article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

machine translation etc. (Mehta and Desai, 2011), developing a tagger becomes an essential step in making a language computational.

Indian languages, especially for Odia, the official language of Odisha the tagging system is in an embryonic stage. The language is spoken by 45 million people including Odisha and some of its neighboring states (Behera, 2015) but in the computational scenario it is dormant and development of an efficient POS tagger is still an unsolved problem. One major issue is the unavailability of quantitative as well as qualitative training data. Secondly its rich morphology (Mohapatra, 2007, Pradhan et al., 2016) makes the work difficult. These prevailing issues are the instinct of motivation to carry out the proposed work. The article makes a positive attempt to solve the existing issue and proposes a tagger that is a judicious amalgamation of linguistic rules and machine learning techniques like HMM and ME. Thus the proposed method is a hybrid method that solves the objective of disambiguating lexical tokens (words) for a morphologically rich language constrained by limited training corpus.

The underlying work also analyses some of the algorithms namely rule based, HMM, modified maximum entropy (ME) and conditional random field (CRF) with their implementation on Odia text. As the performance of the taggers varies in accordance with the input language and provided knowledge base like training corpora, the said methods are exhaustively evaluated and compared with the proposed hybrid method.

The rest of the paper is expanded as follows: second section discusses some of the specific approaches adopted for building a tagger and state of art of Odia language in this scenario. Third and fourth sections give a brief description of the tag set and data set used in the experiment respectively. Fifth section elaborates the behavior of implemented POS taggers. Sixth section presents the proposed hybrid tagger. Seventh section does a result analysis of the hybrid method with respect to other algorithms and finally the work is concluded with a future direction.

RELATED WORKS

This section discusses some of the selective works related to the proposed methodology and the present state of art of taggers developed for Odia. Starting with rule based technique a primitive method has been efficiently adopted in building English tagger (Brill, 1992; Pham, 2020). For Indian language like Hindi (Garg et al., 2012) the said method has also proved efficient by giving a noticeable performance of 87.55% accuracy. But in due course researchers observed that developing taggers monopolized by linguistic rules proved to be a difficult task. A shift towards statistical methods that comply more on axioms of probabilities took place proving relatively more efficient.

HMM an efficient statistical approach has shown a noticeable performance of 92.13% in building a Hindi tagger (Joshi et al., 2013). For Myanmar language (Zin and Thein, 2009) the same approach has been successful in giving 96.56% accuracy. Both supervised and unsupervised learning is done using pre tagged and untagged corpus respectively. Afini and Supriyanto (2017) applied improved HMM with affix tree in the tagger for Indonesian language. Combination of a morphological analyzer named MorphInd and HMM is used to achieve higher performance. Other efficient method like ME experimented by the proposed method was initially approached by Ratnaparkhi (1996) for tagging English words obtaining 96.6% result. A POS tagger (Ma et al., 2010) using ME and hierarchical word clustering for Chinese language was developed. The authors adopted this method to solve the data sparseness problem and have got 93.35% accuracy. CRF similar to ME has proved as an efficient method for grammatical tagging. A CRF based tagger in a Finite state framework was proposed (Constant and Sigogne, 2011), where the decoding was done using weighted finite-State transducer composition. It reports of achieving a promising result of 94% for French language.

Though Indian languages lack behind in the race some of the prominent taggers for Indian languages are rule based POS tagger for Hindi (Garg et al., 2012), hybrid tagger for Hindi (Mohnot et al., 2014) consisting of both rule based and HMM approach with an average accuracy of 89.9%, tagger for Kannada language (BR and Kumar, 2012) using HMM and CRF claiming an accuracy of

79.9% and 84.58% respectively. Taggers for other Indian languages like Bengali, Punjabi, Tamil etc. (Antony et al., 2011) varying in their morphology are also witnessed.

Considering the state of art of POS tagger for Odia language, limited progress is accounted. Taggers are build up on techniques like ANN (Artificial Neural Network), SVM and CRF. A POS tagger (Das and Patnaik, 2014) based on ANN claimed of getting accuracy 81%. The model is implemented on a small size corpus. No clear information about the tag set and the evaluation metric is mentioned. An SVM based Odia POS tagger (Das et al., 2015) reported of obtaining an accuracy of 82% with a small tag set of five tags with a training size of 10,000 words. Another Odia SVM POS tagger (Ojha et al, 2015) framed on BIS tag set with a training size of 90k words and testing size of 2k words has been reported. The model claims an accuracy of 93.6%. Similar attempts have been made with a training corpus obtained from ILCI (Indian language corpora initiative) and testing the model on seen data (test data part of training set) and unseen data (test data exclusive with the training set). Models like SVM and CRF built on this corpus have been reported achieving an improved result in terms of precision and recall (Behera, 2015).

Thus it can be observed from the literature that similar approaches perform differently if the training set varies in language, quantity and quality. Morphology of the language has a deep impact on the efficiency of a tagger. It is also observed that Indian languages especially Odia lags behind in the race as it is devoid of computational resources. With an objective of building a tagger backed with available limited computational resources a hybrid approach is proposed. It aims to give a clear visualization of the adopted method and a detailed analysis of the result.

POS TAG SET

The tag set considered for all the experimented models is with reference to the BIS (The Bureau of Indian Standards) annotation standard (Chandra, N. et al., 2014) and Unified POS tag set for Indian Languages (<http://tdil-dc.in>), considering only the higher level POS class. In the corpus linguistic lore the structure of the tag set and the number of tags depends on the language structure and the available training corpus. If the training size is small the level of granularity in the tag set is generally coarse. The tags higher in the hierarchy i.e. the major tags which are sufficient to provide the grammatical structure of the sentence are considered. This approach gives a less ambiguous result and the tagger can be erroneously applied for other text processing systems. Table 1 gives a clear view of the considered tags with their abbreviation.

DATA SET EXPERIMENTED

The text corpus considered for the experiments belong to tourism domain. It has been obtained from TDIL (Technology Development for Indian Languages, govt. of India) (<http://tdil-dc.in>). It is a program started by Govt. of India for language development and maintenance like developing corpus for computational works. The corpus has been developed under ILCI (Indian language corpora initiative) project by annotators at different phases (Jha, 2012). The corpus size counts to about 310,290 words or tokens. It is mainly noun and verb dominated. Figure 1 shows the overall statistic of the distribution of tags in the training set. For training and testing purpose the corpus size of 80% and 20% is considered respectively. The models performance has also been tested on corpus belonging to varied domains like cricket, health, railways etc. collected from news articles and annotated by expert linguistics working on Odia language. The purpose of taking words from different domains is to have a better quality evaluation of the tagger.

The training data has been mapped to a format suitable for our experiment.

e.g.

Original data set format

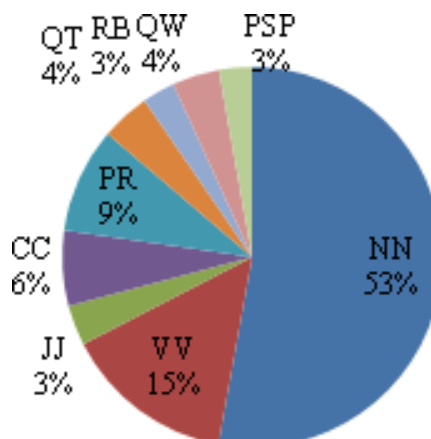
Table 1. POS tags considered

POS Category	Label	Example	Transliteration	Gloss
Noun	NN	ଫୁଲ, ପୁରି	<u>phula</u> , <u>puri</u>	flower, Puri
Pronoun	PR	ଆମେ, ଆମର	<u>āme</u> , <u>āmara</u>	we, our
Verb	VV	ଖାଉଛନ୍ତି	<u>khāuchhi</u>	eating
Adjective	JJ	ସୁନ୍ଦର	<u>sundara</u>	beautiful
Adverb	RB	ସିଘ୍ର	<u>sighra</u>	fast
Postposition	PSP	ସାମ୍ନାରେ	<u>sāmnāre</u>	in front of
Particle	RP	ନ, ନୁହେଁ	<u>na</u> , <u>nuhē</u>	Negative words, words having an association with verbs and are not lexically well defined.
Conjunction	CC	ଓ, ମଧ୍ୟ	<u>o</u> , <u>madhva</u>	<u>and</u> , also
Quantifier	QT	୧, ୩, ସାତଟା	<u>1</u> , <u>3</u> , <u>sātata</u>	1, 3, seven
<u>Question words</u>	QW	କାହାର, କେବେ	<u>kāhāra</u> , <u>kebe</u>	whose, when
Punctuation	PUNC	।, \ ? () { }	। (Full stop .) Others have same representation	

htd23008 ଚାମୁଣ୍ଡା\N_NNP ଦେବୀଙ୍କର\N_NN ମନ୍ଦିର\N_NN ବାଗିଚା\N_NNP ତଟରେ\N_NN ଅବସ୍ଥିତ\N_NN | \RD_PUNC
Transliteration: htd23008 chāmunda\N_NNP debīṅkara\N_NN mandira\N_NN bāgaṅgāra\N_NNP taṭare\N_NN abasthita\N_NN | \RD_PUNC
Mapped data set format

<ST> ଚାମୁଣ୍ଡା_NN ଦେବୀଙ୍କର_NN ମନ୍ଦିର_NN ବାଗିଚା_NN ତଟରେ_NN ଅବସ୍ଥିତ_VV |_PUNC <END>

Figure 1. Tag distribution in the training corpus



Transliteration: <ST> chāmūṇḍā_NN debiṅkara_NN mandira_NN
bāṇagaṅgāra_NN taṭare_NN abasthita_VV |_PUNC <END>

Translation: The temple of Chamunda Devi is located at the
bank of Banganga.

Where, N_NNP (Proper noun) and N_NN (Common noun) is mapped to NN
(Noun), RD_PUNC (Punctuation under residual tag category) to PUNC
(Punctuation) and V_VM (main verb) to VV.

htd 23008 is sentence index in the original format and <ST> and <END> are start and end of
a sentence in the mapped corpus file respectively.

POS TAGGER FOR ODIA TEXT BUILD ON VARIED APPROACHES

This section discusses the implementation of the tagger with few trusted and efficient approaches
for a comparative analysis with the proposed methodology with respect to Odia text. The analysis is
done with an aim to propose a robust and efficient POS tagger.

Odia Rule Based POS Tagger

Rule based models are language dependent requiring deep linguistic knowledge. A tagger built on
derived rule set and a lookup table L (Appendix) is evaluated for the considered Odia data set. The
lookup table comprises of lists for punctuation, postposition, conjunction, question word, pronoun,
suffixes and prefixes. The rules are listed with the help of linguists who are acknowledged in the
acknowledgement section. Algorithm1 gives a systematic flow of steps followed.

Algorithm 1. Rule based tagging process

Input: Set of untagged test sentences $S_{in}=\{s_1, s_2, \dots, s_n\}$ and a lookup
table L .

Output: Set of tagged sentences S_{out}

For each sentence s_i :

do

 Tokenize s_i into tokens

 For each token w_j in s_i

 do

 perform a look up into the lookup table L

 compute suffix and prefix matching

 derive Set of potential tags

 If single tag found:

 tag it to the token

 Else If multiple tags are found associated with the token:

 resolve the ambiguity using set of derived rules and

associate suitable tag to the token

 Else (no tag found):

 associate the token with tag noun

 done

 Detokenize and display the final set of tagged sentences S_{out}

done

The confusion matrix of table 2 depicts the tagging accuracy of different tags after the testing
process and also their % mistag. The zero entries in the table signify that a particular tag in a
row is not mistagged with the corresponding tag in the column. It can be visualized that percentage
accuracy of tag particle (RP) obtained is lowest as there is lack of standard rules and suffixes contrary
to noun. The accuracy percentage in case of adjective and adverb could not reach a higher value due
to lack of sufficient disambiguation rules. Noun tag, pronoun tag, verb tag and punctuation tag has got

Table 2. POS Confusion Matrix in Rule Based Model (%)

	NN	VV	JJ	CC	PR	QT	QW	RP	RB	PUNC	PSP
NN	93.54	2.19	1.73	0.205	1.12	0.028	0.019	0.0312	0.053	0	1.08
VV	31.3	67.65	0.067	0.016	0.052	0	0	0	0.076	0	0.825
JJ	52.23	0.775	45.76	0.057	0.88	0	0	0.008	0.008	0	0.277
CC	4.19	0.645	0.046	84.52	8.078	0	0.084	0	0.077	0	2.352
PR	10.77	0.258	0.376	0.869	84.55	0.006	1.39	0	0.074	0	1.72
QT	38.34	6.567	2.79	0.02	3.15	48.91	0.071	0	0	0	0
QW	0	0	0	1.275	54.425	0	44.3	0	0	0	0
RP	47.47	8.35	0.559	8.84	4.29	0.47	0	28.453	0	0	0.017
RB	33.77	0.166	1.48	0.328	37.23	0	0	0	34.28	0	20.92
PUNC	0	0	0	0	0	0	0	0	0	100	0
PSP	22.64	0.014	0.216	1.587	3.03	0	0	0	1.833	0	70.669

appreciable accuracy percentage compared to other tags as they are supported by sufficient rules and a lookup table. Similarly conjunction and postposition tag have also got good accuracy percentage. Thus the confusion matrix shows the performance of the method.

HMM Based Odia POS Tagger

As deriving rules is a cumbersome task and need regular updates, the statistical bigram model HMM was experimented for building Odia tagger. It is generative in nature and the system being modeled is assumed to be a Markov process (Kupiec, 1992).

The two assumptions made by HMM are:

1. The probability of a word or token appearing depends only on its own tag, not on neighboring tags or words i.e.

Emission probability

$$P(w_1^n / t_1^n) \approx \prod_{i=1}^n P(w_i / t_i) \quad (1)$$

Where w is the token and t is the associated tag and i varies from 1 to n the length of the sequence.

$P(w_i / t_i)$ gives the probability of a tag t_i to be associated with a token w_i .

A bigram assumption i.e. the probability of the tag is dependent only on the previous tag, not on the entire tag sequence i.e.

Transition Probability

$$P(t_1^n) \approx \prod_{i=1}^n P(t_i / t_{i-1}) \quad (2)$$

Where t_i is the current tag and t_{i-1} is the previous tag.

Thus we find the best tag sequence (T^*) given a word sequence, as a combination of emission and transition probabilities.

$$T^* = \operatorname{argmax} P(T / W) \quad (3)$$

$$\approx \operatorname{argmax} \prod_{i=1}^n P(w_i / t_i) P(t_i / t_{i-1})$$

To check the exponential growth of the paths dynamic Viterbi algorithm combines with basic HMM and finds the best path without enumerating all paths explicitly. The computation applies equation 4 to get the Viterbi entries $V(t,i)$ (Cahyani and Vindiyanto, 2019).

$$V(t, i) = \max V(t_{i-1}, i-1) \cdot P(t_i, t_{i-1}) \cdot P(w_i, t_i) \quad (4)$$

80% of the TDIL corpus is considered for training and 20% is considered for testing. Table 3 shows the performance result trained on the considered Odia data set.

From table 3 it can be visualized that the experimented HMM approach is showing an appreciable performance even with a small size corpus. The tagging accuracy for adjective (JJ) and adverb (RB) tags are poor. The reason is their ambiguous nature which can be seen from the cited example. Some of the mistags are also due to errors in training data.

Table 3. POS Confusion Matrix for HMM POS Model (%)

	NN	VV	JJ	CC	PR	QT	QW	RP	RB	PUNC	PSP
NN	97.03	0.43	1.41	0.13	0.25	0.23	0	0.16	0.027	0	0.2
VV	11.64	87.53	0.5	0	0.025	0	0	0.07	0	0	0.21
JJ	49.68	4.05	44.73	0	0.59	0.387	0	0.455	0	0	0.091
CC	3.396	0.133	0.033	93.93	1.13	0.16	0	0.63	0	0	0.57
PR	2.95	0	0	1.97	93.34	0	0.6	0.09	0.03	0	1.01
QT	7.80	0	2.48	0.155	0.62	85.94	0	2.98	0	0	0
QW	0	0	0	0	25	0	75	0	0	0	0
RP	11.43	1.056	3.66	8.07	1.8	1.61	0	67.308	0.124	0	4.909
RB	26.08	4.34	4.34	0	0	0	0	4.34	47.82	0	13.04
PUNC	0.04	0	0	0	0	0	0	0	0	100	0
PSP	12.04	0.2	0	0.3	1.31	0	0	0.4	0.2	0	85.56

e.g. ସମସ୍ତଙ୍କୁ_NN ଆକର୍ଷିତ_NN/JJ କରୁଥାଏ_VV

samasta_ku_NN ākarsita_NN/JJ karithāe_VV

Here (ākarsita) shows an ambiguous nature between noun and adjective.

As Odia is morphologically rich (Pattnaik and Nayak, 2020) the model should be approached with feature based methodologies to have a performance comparison and maximum entropy concept exploits the features effectively in POS tag prediction (Ekbal et al., 2008; Ratnaparkhi, 1996). Thus these concepts are further experimented in the article on Odia language.

A Modified ME Based Odia POS Tagger

ME model is a feature based language model. Varied forms of contextual information have been combined in a meticulous manner. The objective of the model is to find T^* (equation 3) the most probable tag sequence for a given word sequence.

The probability model takes the space $H \times T$.

H is the set of contexts in which a word appears and T is the tag sequence.

Maximum entropy approach specifies a set of features from the environment for tag prediction.

A random variable h represents these features and t represents the POS tag.

$h_i = \{w_i, w_{i+1}, w_{i-1}, t_{i-1}, t_{i-2}\}$ is an environment for a token w_i .

The tagger learns a log linear conditional probability model from tagged text.

The probability distributions having the highest entropy out of those distributions and satisfying a certain set of constraints is considered (Ratnaparkhi, 1996) and forced to obey the rule:

$$Ef_j = \tilde{E}f_j, \quad 1 \leq j \leq k \quad (5)$$

Where,

$$Ef_j = \sum_{h \in H, t \in T} p(h, t) f_j(h, t) \quad (6)$$

model's feature expectation.

$\tilde{E}f_j$ is the observed feature expectation in the training data

$$\tilde{E}f_j = \sum_{i=1 \text{ to } n} \tilde{p}(h_i, t_i) f_j(h_i, t_i) \quad (7)$$

$\tilde{p}(h_i, t_i)$ is the observed probability of (h_i, t_i) in the training data.

$f_j(h_i, t_i)$ is the feature function of the ME model.

These features set the relations between the contextual factors h and the POS tag t i.e.

$$f_j(h_i, t_i) = \begin{cases} 1 & \text{if } \{h, t\} \text{ satisfying the condition} \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

Table 4. Maximum Entropy Feature Template

	Feature template
$\forall w_i$	$W_i=x, t_i=TG$
	$t_{i-1}=u, t_i=TG$
	suffix $t_i=TG$
	prefix $t_i=TG$
	$W_{i-1}=x, t_i=TG$
	$W_{i+1}=x, t_i=TG$
	$t_{i+1}=END, t_i=TG$

Thus a feature function takes either a value 0 or 1

$$p(h, t) \approx \tilde{p}(h) \cdot p(t / h) \quad (9)$$

$$Ef_j \approx \sum_{i=1 \dots n} \tilde{p}(h_i) p(t_i / h_i) f_j(h_i, t_i) \quad (10)$$

Thus equation 6 (model's feature expectation) takes the form of equation 10.

$p(t / h)$ is the conditional distribution.

$\tilde{p}(h_i)$ is the observed probability of the history h_i in the training set .

$$p(t / h) = \frac{\sum e^{\lambda_j f_j(h, t)}}{z(x)} \quad (11)$$

$$z(x) = \sum_{t' \in T} \prod_{j=1 \dots k} e^{\lambda_j f_j(h, t')} \quad (12)$$

The denominator is as a factor for normalization.

λ_j is the weight for a certain feature derived from the training set through Generalized Iterative scaling (GIS) (Wilhelm et al., 2018).

The feature template designed for the experiment is shown in table 4.

Where w_i is any token that carries a value x and u is any tag that the variables t_i can carry.

TG is the tag predicted on the existence of a feature.

A modification is made in the suffix and prefix part of the designed template, they are predefined with the help of expert linguists. A set of 207 suffixes and 21 prefixes (Appendix) are used in the process. This leads to faster execution preventing the model from missing any suffixes and prefixes important in analyzing Odia word, ignoring the prevailing fact of limited training corpus.

Algorithm 2 gives the steps followed by the maximum entropy approach for Odia POS tagger.

Table 5. POS Confusion Matrix for Odia ME model (%)

	NN	VV	JJ	CC	PR	QT	QW	RP	RB	PUNC	PSP
NN	92.85	3.378	1.546	0.16	1.17	0.297	0	0.198	0.015	0	0.377
VV	7.335	89.95	2.28	0.088	0.076	0.038	0	0.013	0	0	0.215
JJ	51.23	2.620	44.37	0.068	0.843	0.524	0	0.273	0.023	0	0.045
CC	5.461	0.632	0.066	89.81	3.16	0.166	0	0.1	0	0	0.6
PR	8.385	0.238	0.059	0.686	89.88	0	0.626	0.059	0.03	0	0.03
QT	9.55	0.62	0.7	0.11	0.66	87.42	0	0.93	0	0	0
QW	0	0	0	0	25	0	75	0	0	0	0
RP	18.33	2.237	0.807	7.4	6.21	19.2	0	44.87	0.124	0	0.808
RB	34.78	0	4.35	4.35	8.69	0	0	4.35	43.47	0	0
PUNC	0	0	0	0	0	0	0	0	0	100	0
PSP	27.56	0.93	0	0.146	9.75	0	0	0	0	0	61.61

Algorithm 2: ME adopted for Odia POS Tagger

Training Phase:

Input annotated corpus and customized feature template.

Create a database with context information for each token.

Derive feature functions with their weights using Generalized Iterative scaling (GIS).

Testing phase:

Test sentence: $W = \{w_1, w_2, \dots, w_n\}$

TS_{ij} : j th highest probability tag sequence for w_i .

$\tilde{p}(f_i) > count$: $\tilde{p}(f_i)$ is the probability of occurrence of a feature in the training set.

count = 2

K=2 is the beam size

Input: Training corpus

Test corpus

Generate tags for w_i using equation 9

Proceed for $\tilde{p}(f_i) > count$.

Find top K

Set TS_{ij} , $1 \leq j \leq K$ accordingly.

Initialize $i = 2$

do

Initialize $j = 1$

do

Given $TS_{(i-1)j}$, generate tags for w_i and append it to $TS_{(i-1)j}$ to make a new sequence.

$j = j + 1$; repeat if $j \leq K$

done

From the above loop find K highest probability tag sequences.

Set TS_{ij} , $1 \leq j \leq K$

$i = i + 1$, repeat if $i \leq n$.

done

Return highest probability sequence TSn_1 .

Table 5 shows the performance result after the model is run on the considered data set.

It can be visualized from the confusion matrix the tag noun has obtained maximum accuracy. For tags adjective, adverb and particle ambiguity level is high which the method couldn't resolve due to limited training corpus. Though ME approach efficiently utilizes the morphological features for the tag prediction, lack of sufficient training data leads to misprediction of suffixes and erroneous tagging could not meet the expected performance.

CRF Based Odia POS Tagger

It is a popular discriminative model that considers multiple features for structured prediction. It has all the advantages of ME without the label bias problem and is defined (Sutton & McCallum, 2011) on observed feature vector X (sequence of tokens in a sentence) producing an output vector of random variables Y (POS tags) i.e.

$$P(Y / X) = \frac{1}{Z(X)} \prod_{t=1}^T \exp \left\{ \sum_{k=1}^K \theta_k f_k(y_t, y_{t-1}, x_t) \right\} \quad (13)$$

$X = \{x_1, x_2, \dots, x_n\}$ is the input sequence.

$Y = \{y_1, y_2, \dots, y_n\}$ is the output sequence of labels.

$Z(X)$ is the normalization factor.

It is the sum of all possible state sequences summing to 1.

f_k a binary feature

θ_k the weight of the binary feature.

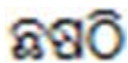
y_t is the label of x at the current position t .

Thus CRF has a similar approach as maximum entropy algorithm with a difference in normalization. ME adopts local variance normalization whereas CRF adopts global variance normalization. This normalization process overcomes the label bias problem of ME.

Table 6 gives the confusion matrix of the result obtained.

Though CRF is giving higher performance than ME, still it faces the problem of insufficient training data. Unavailability of suitable benchmark data set and error in file leads to performance degradation. Though the files have been rectified with the help of linguists still some ambiguity exists. Low and erroneous training data leads to misinterpretation of suffixes.

e.g.



(chhasaକ୍ଷୀhi) is tagged as PR instead of Quantifier (QT) by the model as frequency of (କ୍ଷୀhi) suffix is 2 for noun (NN), 8 for pronoun (PR) and 1 for verb (VV).

After evaluation of the generated outputs of the discussed methods and their effectiveness on Odia text the motivation towards hybridized approach that could integrate the positive features of

Table 6. POS Confusion Matrix for Odia CRF model (%)

	NN	VV	JJ	CC	PR	QT	QW	RP	RB	PUNC	PSP
NN	95.839	1.51	1.482	0.256	0.315	0.170	0	0.11	0	0	0.31
VV	9.565	87.7	2.21	0.063	0.114	0.012	0	0.013	0.05	0	0.29
JJ	55.69	0.88	41.81	0.022	0.638	0.524	0	0.342	0	0	0.07
CC	2.644	0.36	0.066	93.27	2.664	0.166	0	0.299	0	0	0.49
PR	7.997	0.15	0.059	0.298	90.71	0	0.059	0.119	0	0	0.59
QT	12.189	0.35	0.621	0.077	0.427	84.62	0	1.669	0	0	0.04
QW	0	0	0	0	25	0	75	0	0	0	0
RP	18.955	1.80	0.621	6.215	1.243	6.898	0	63.20	0	0	1.06
RB	30.43	0	13.04	0	0	0	0	8.695	43.5	0	4.35
PUNC	0	0	0	0	0	0	0	0	0	100	0
PSP	20.73	0.39	0	0.146	0.243	0	0	1.463	0	0	77.024

the previously discussed concepts and enhance the obtained result stemmed. The following proposed tagger is a combination of stochastic and linguistic features.

PROPOSED HYBRID ODIA POS TAGGER

The proposed method adopts a hybrid structure keeping into consideration the constraint of having small size Odia training corpora. It is a combination of statistical and rule based method. Unknown words are tagged using a lookup table, derived rules and the modified ME algorithm. All these processes are articulated in a clean way. Varied forms of contextual information have been combined in a systematic manner. Thus it is a hidden Markov based model that passes through a Viterbi path and efficiently utilizes the morphology of the language through maximum entropy to get an output sentence with each word tagged with its appropriate part of speech.

Algorithm 3 gives the overall procedure followed by the proposed tagger.

Algorithm 3: Adopted approach for the hybrid tagger

Training Phase:

Input tagged corpus

Generate the emission and transition probability sets for each word w_i in the training set.

Emission Probability: $P(w_i / t_i)$

Transition Probability: $P(t_i / t_{i-1})$

Testing Phase:

Input:

Set of untagged sentences $S = \{s_1, s_2, s_3, \dots, s_j\}$

$P(w_i / t_i)$ a set of derived tags for w_i

$P(t_i / t_{i-1})$ a set of previous tag pairs for t_i

Initialize $V(t, i) = 1$

For each sentence s_j in the test sentence:

do

 Tokenize the sentence into words w_i .

```

    For each  $w_i$  in  $s_j$  find the best path of length  $i-1$  to each
    state;
    do
        If  $w_i \in$  training set:
            Generate the set of values  $P(w_i / t_i)$  and  $P(t_i / t_{i-1})$  from
            the training set.
        Else:
            Go to algorithm 4 and compute  $P(w_i / t_i)$ .
    Compute and store the partial results  $V(t, i)$  as the path proceeds
    according to equation 4.
    Consider the maximum value over each possible previous tag  $t_{i-1}$ .
    Proceed from left to right of the test sentence.
    Choose the best path from the available computed paths to reach
    the final state.
    Store a back trace to show from which state ( $i-1$ ) state ( $i$ ) came
    from.
    Return tag sequence ( $T^*$ ) for the given word sequence by back
    tracing.
done
done

```

Algorithm 4 describes the step wise procedure followed to tag unknown words.

Algorithm 4: Tagging unknown words

```

For each unknown word  $w_i$ 
    if  $w_i$  found in the lookup table
        resolve the ambiguity using rule based
        algorithm (algorithm 1)
         $P(w_i / t_i) = \text{Average probability of } t_i \text{ in the training corpus}$ 
    Else if
         $w_i \in$  Numeric data,  $t_i = \text{QT(Quantifier)}$ 
         $P(w_i / t_i) = \text{Average probability of } t_i \text{ in the training corpus}$ 
    Else
        Find the probability of  $w_i$  using modified ME (algorithm 5)

```

Modified ME template (Table 7) used for tagging unknown words by the hybrid method is similar to table 4 except it lacks the feature template ($w_i = x$ and $t_i = \text{TG}$).

The nomenclatures are in accordance with modified ME based Odia POS tagger section.

Table 7. Template for solving unknown words

Words	Features
$\forall w_i$	$t_{i-1} = Z, t_i = \text{TG}$
	Suffix, $t_i = \text{TG}$
	Prefix, $t_i = \text{TG}$
	$w_{i-1} = x, t_i = \text{TG}$
	$w_{i+1} = x, t_i = \text{TG}$
	$t_{i+1} = \text{END}, t_i = \text{TG}$

Algorithm 5 describes the step wise procedure followed to tag unknown words through ME.

Algorithm 5: ME modified for handling unknown words

Derive the features of the words according to the template in table 7.

Determine the weight λ_j for each derived feature using Generalized Iterative Scaling.

Calculate the probability of word w_i to be tagged with tags t_i with the existing features according to the equation 11 and 12. The tag that takes the maximum probability of a given token is assigned.

The feature weights λ_j are obtained by Generalized Iterative Scaling.

K fold cross validation has been conducted to validate the process.

This is an essential step for a machine learning system mainly where data size is limited. It ensures that every observation from the original data set has a chance of appearing in the training and testing set (Jiang & Wang, 2017). The value of K is kept at 4.

Table 8 gives a clear view of the accuracy % of the tagger for four iterations and the overall variance that exist among them.

This low value of variance i.e. 0.19 signifies the stability of the method.

Table 9, the confusion matrix of the hybrid model shows an improvement in the result.

The performance of noun, verb, postposition and conjunction are high. But comparatively the model functions less accurately for adjective and adverb. Some of the reasons are:

Table 8. Performance of the tagger at each iteration

	Overall % accuracy	Variance
Iteration 1	93.72	0.19
Iteration 2	94.03	
Iteration 3	93.082	
Iteration 4	92.874	

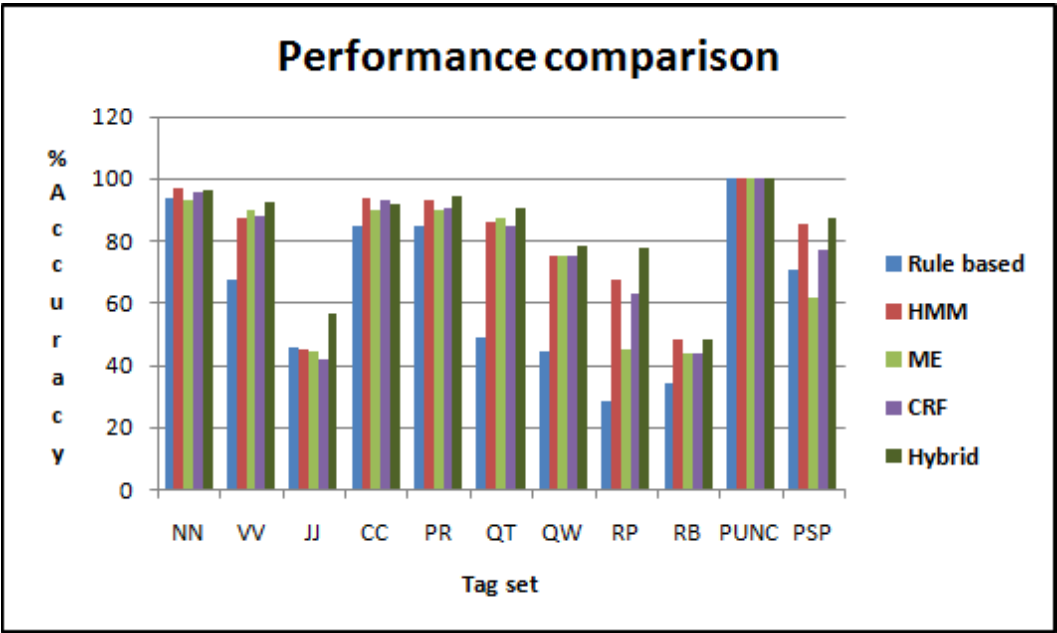
Table 9. POS Confusion Matrix for proposed hybrid model (%)

	NN	VV	JJ	CC	PR	QT	QW	RP	RB	PUNC	PSP
NN	95.86	1.344	1.320	0.1023	0.464	0.368	0	0.301	0.017	0	0.224
VV	7.56	91.208	0.992	0.042	0.0422	0	0	0.137	0	0	0.021
JJ	27.68	0.340	68.235	0	0.4537	0.851	0	2.38	0.056	0	0
CC	1.425	0.048	0	90.698	2.657	0.532	0	0.314	0	0	4.325
PR	2.139	0	0.0517	0.483	94.912	0.034	0.15	0.155	0.0172	0	2.052
QT	5.71	0.148	0.63	0.1112	0.1483	92.92	0	0.334	0	0	0
QW	0	0	0	0	22.38	0	77.6	0	0	0	0
RP	6.987	0.056	5.534	0.5589	0.0559	1.621	0	85.18	0	0	0
RB	14	0	7.24	0	6.28	0	0	0	53.14	0	19.32
PUNC	0	0	0	0	0	0	0	0	0	100	0
PSP	5.62	0.118	0.06	1.065	1.301	0	0	6.68	0	0	85.09

1. Words belonging to adjective and adverb are very ambiguous.
2. Lack of sufficient benchmark error free training corpus.
3. Lack of sufficient rules for disambiguation.

The most frequent error observed is the miss tagging of adjective as noun. This was due to the

Figure 2. Comparative analysis of accuracies of different methods



order of occurrence of adjectives in the noun phrases (Kupiec, 1992). But compared to other models miss tagging of adjectives is reduced and can be visualized from Figure 2.

Thus a comparison of methods on the basis of accuracy percentage of each tag is depicted in Figure 2.

Table 10 shows the accuracy % obtained in terms of F score for individual tag by each method.

The methods have also been tested on text belonging to other domain apart from tourism i.e. on Odia news text covering data from varied domains. Table 11 shows their performance.

It can be observed that the hybrid approach is performing better in comparison to the rest of the approaches in case of other domain data set.

RESULT DISCUSSION

A comparative analysis of the results obtained by the implemented approaches on Odia text (Figure 2 and Table 10) shows the dominance of proposed hybrid approach on rest of the methods obtaining highest accuracy of 94.6% F score for noun tag. The tagging error for the tags verb, postposition, conjunction and pronoun by all the implemented methods are low as these tags are comparatively less ambiguous. Adjective, adverb, particle and question word have maximum tagging errors comparative to other tags. It can also be visualized that machine learning approaches compared to rule based have been successful in decreasing these tagging errors to some extent. For example 43.77% accuracy

Table 10. Comparison of F score among models

Tags	F score of different Models in %				
	Rule based	Max Entropy	CRF	HMM	Hybrid
NN	85.74	90.05	92.66	92.7	94.60
VV	72.25	87.12	89.85	91.09	92.96
JJ	30.08	55.218	53.12	56.2	63.88
CC	88.14	91.25	94.02	94.31	94.85
PR	81.73	83.65	91.67	93.38	94
QT	64.54	85.53	88.06	90.10	92.86
QW	38.58	41.38	66.66	70.58	77.32
RP	43.77	59.20	74.09	75.26	78.75
RB	42.98	58.823	54.05	56.41	59.59
PUNC	100	100	100	100	100
PSP	68.59	72.46	83.58	85.59	87.02

Table 11. Comparison of % accuracy among methods for other domain

Models	% Accuracy
Rule based	83.39
Maximum Entropy	84.07
CRF	85.02
HMM	86.9
Hybrid	89.09

obtained on the basis of F score for the tag particle by rule based is seen to be enhanced by ME to 59.20%, further by CRF to 74.09%, by HMM to 75.26% and to maximum by hybrid approach to 78.75%. Still rules cannot be ignored. Though stand alone rule based models donot excel in performance but their integration with other methods enhances the efficiency and is proved by the experimented proposed hybrid method. From table 11 it can also be seen that the proposed model when applied on other domain like texts from news corpus related to cricket, health and railways is also showing better performance of 89.09%. Overall the proposed hybrid approach trained on limited size corpus belonging to tourism domain is showing an enhanced result 94.03% on unseen data (data not a part of the training set) compared to previous state of art.

CONCLUSION AND FUTURE WORK

This paper has proposed a hybrid approach for POS tagging with a comparative analysis done with other methods that are rule based, HMM, ME and CRF. The proposed architecture can be considered as an innovative work that has exploited the complex morphology of Odia texts to develop a suitable POS tagger. Compared to the present state of art of Odia POS taggers, the proposed work gives a better performance undergoing proper evaluation phase with a transparent view of the whole process. Every language is exclusive with its unique grammatical construction, so the knowledge of the linguists

hybridized with sophisticated machine learning technique has made the tagger performing. The obtained score of 94.03% accuracy by the proposed hybrid approach with the limited training corpus can be considered as a noted work. The technique adopted shows scope for research in development of taggers lacking voluminous corpora more specifically for morphologically complex Indian languages on which presently available tools cannot be directly applied. The proposed technique can be applied to other similar languages by making few modifications at the linguistic level. The few discrepancies and errors existing in the output affecting the accuracy can be removed by increasing the training size and appending more linguistic rules. This can be considered as a future work. Conclusively the work can be considered as a significant contribution in the development of a morphologically rich language in the computational scenario. Further it is intended to examine the potentiality of the proposed POS tagger in the development of automatic text summarizer for Odia text documents.

ACKNOWLEDGMENT

As the domain of the work includes both linguistics and computer science, so we have taken help from linguists, grammarians and organizations working in Odia language. In this regard we acknowledge

the support extended by team Srujanika, Bhubaneswar and Dr. Hare Krishna Patra of Kedarnath Gabesana Pratisthana, Bhubaneswar.

REFERENCES

- Afini, U., & Supriyanto, C. (2017, November). Morphology analysis for Hidden Markov Model based Indonesian part-of-speech tagger. In *2017 1st International Conference on Informatics and Computational Sciences (ICICoS)* (pp. 237-240). IEEE.
- Antony, P. J., & Soman, K. P. (2011). Parts of speech tagging for Indian languages: a literature survey. *International Journal of Computer Applications*, 34(8).
- Behera, P. (2015). *Odia parts of speech tagging corpora: suitability of statistical models* (Doctoral dissertation (M. Phil. Dissertation). Jawaharlal Nehru University (JNU), New Delhi, India.
- Br, S., & Kumar P, R.BR. (2012). Kannada part-of-speech tagging with probabilistic classifiers. *International Journal of Computers and Applications*, 48(17), 26–30. doi:10.5120/7442-0452
- Brill, E. (1992). A simple rule-based part of speech tagger. In *Proceedings of the workshop on Speech and Natural Language*. Association for Computational Linguistics.
- Cahyani, D. E., & Vindiyanto, M. J. (2019, November). Indonesian Part of Speech Tagging Using Hidden Markov Model–Ngram & Viterbi. In *2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)* (pp. 353-358). IEEE.
- Chandra, N., Kumawat, S., & Srivastava, V. (2014). Various tagsets for Indian languages and their performance in part of speech tagging. *Proceedings of 5th IRF International Conference*.
- Constant, M., & Sigogne, A. (2011). MWU-aware part-of-speech tagging with a CRF model and lexical resources. In *Proceedings of the workshop on multiword expressions: from parsing and generation to the real world*. Association for Computational Linguistics.
- Das, B. R., & Patnaik, S. (2014). A novel approach for Odia part of speech tagging using artificial neural network. In *Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2013* (pp. 147-154). Springer. doi:10.1007/978-3-319-02931-3_18
- Das, B. R., Sahoo, S., Panda, C. S., & Patnaik, S. (2015). Part of speech tagging in odia using support vector machine. *Procedia Computer Science*, 48, 507–512. doi:10.1016/j.procs.2015.04.127
- Ekbali, A., Haque, R., & Bandyopadhyay, S. (2008). Maximum entropy based Bengali part of speech tagging. *Advances in Natural Language Processing and Applications, Research in Computing Science (RCS) Journal*, 33, 67–78.
- Garg, N., Goyal, V., & Preet, S. (2012, December). Rule based Hindi part of speech tagger. In *Proceedings of COLING 2012: Demonstration Papers* (pp. 163-174). Academic Press.
- Indian Language Technology Proliferation and Deployment Centre. (n.d.). <http://tdil-dc.in>
- Jha, G. N. (2012). The TDIL program and the Indian language corpora initiative. *Language Resources and Evaluation Conference*.
- Jiang, G., & Wang, W. (2017). Error estimation based on variance analysis of k-fold cross-validation. *Pattern Recognition*, 69, 94–106. doi:10.1016/j.patcog.2017.03.025
- Joshi, N., Darbari, H., & Mathur, I. (2013). HMM based POS tagger for Hindi. *Proceeding of 2013 international conference on artificial intelligence, soft computing (AISC-2013)*.
- Kanakaraddi, S. G., & Nandyal, S. S. (2018). Survey on parts of speech tagger techniques. *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*, 1-6. doi:10.1109/ICCTCT.2018.8550884
- Kupiec, J. (1992). Robust part-of-speech tagging using a hidden Markov model. *Computer Speech & Language*, 6(3), 225–242. doi:10.1016/0885-2308(92)90019-Z

Ma, J., Huang, D., & Li, Z. (2010). Chinese POS tagging employing maximum entropy and word clustering. *Journal of Information and Computational Science*, 7(12), 2420–2428.

Mehta, D. N., & Desai, N. (2011). A survey on part-of-speech tagging of Indian languages. *1st International Conference on Computing, Communication, Electrical, Electronics, Devices and Signal Processing*, 34.

Mohapatra, B. P. (2007). *Prachalita, Odia bhasara eka*. Academic Press.

Mohnot, K., Bansal, N., Singh, S. P., & Kumar, A. (2014). Hybrid approach for Part of Speech Tagger for Hindi language. *International Journal of Computer Technology and Electronics Engineering*, 4(1), 25–30.

Ojha, A. K., Behera, P., Singh, S., & Jha, G. N. (2015). Training & evaluation of POS taggers in Indo-Aryan languages: a case of Hindi, Odia and Bhojpuri. In *Proceedings of 7th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics* (pp. 524-529). Academic Press.

Pattanaik, S., & Nayak, A. K. (2020). An Automatic Summarizer for a Low-Resourced Language. In *Advanced Computing and Intelligent Engineering* (pp. 285–295). Springer. doi:10.1007/978-981-15-1081-6_24

Pham, B. (2020). *Parts of Speech Tagging: Rule-Based*. Digital Commons at Harrisburg University.

Pradhan, K. C., Hota, B. K., & Pradhan, B. (2016). *Saraswata Byabaharika Odia Byakarana*. SatyaNarayan Book.

Ratnaparkhi, A. (1996). A maximum entropy model for part-of-speech tagging. *Proceedings of the Conference on Empirical methods in Natural Language Processing*, 1.

Sutton, C., & McCallum, A. (2011). An introduction to conditional random fields. *Machine Learning*, 4(4), 267–373. doi:10.1561/22000000013

Tiwar, U. S., & Siddiqui, T. (2008). *Natural language processing and information retrieval*. Oxford University Press, Inc.

Wilhelm, M., Kern-Isberner, G., Finthammer, M., & Beierle, C. (2018, May). A generalized iterative scaling algorithm for maximum entropy model computations respecting probabilistic independencies. In *International Symposium on Foundations of Information and Knowledge Systems* (pp. 379-399). Springer. doi:10.1007/978-3-319-90050-6_21

Zin, K. K., & Thein, N. L. (2009, December). Part of speech tagging for Myanmar using hidden Markov model. In *2009 International Conference on the Current Trends in Information Technology (CTIT)* (pp. 1-6). IEEE. doi:10.1109/CTIT.2009.5423133

APPENDIX A

The suffix and prefix list has been prepared with the help of expert linguists.

Suffix list

"ଗାଏ", "ଟିଏ", "ଗା", "ଟି", "ଟେ", "ମାନ", "ମାନେ", "ଗୁଡ଼ାଏ", "ଗୁଡ଼ିଏ", "ଗୁଡ଼ାକୁ", "ଗୁଡ଼ିକୁ", "ଗୁଡ଼ିଏ", "କୁ", "କୁ", "କି", "କ", "ଗାକୁ", "ଟିକୁ", "ଠାକୁ", "ଠିକୁ", "ଠିକି", "ମାନକୁ", "ରେ", "ହାରା", "ଦେଇ", "କହାରା" etc.

Transliteration:

"āe", "ie", "ā", "i", "e", "māna", "māne", "gurāka", "gurika", "gurāku", "guri", "gurie", "ku", "ku", "ki", "ka", "āku", "iku", "hāku", "hiku", "hiki", "mānaku", "re", "duārā", "dei", "kaduārā", etc.

Gloss: The suffixes donot have an equivalent translation in English. These are the morphemes without having a definite meaning and are not stand alone.

Prefix List

"ପ୍ର", "ପରା", "ଅପ", "ସମ୍", "ନି", "ଅଧି", "ସୁ", "ନି", "ଉତ୍", "ପରି", "ଅବ", "ଅନ୍ତ", etc.

Transliteration

"pra", "parā", "apa", "sam", "ni", "adhi", "su", "ni", "ut", "pari", "aba", "anu", etc.

Gloss: The prefixes do not have an equivalent translation in English. These are the morphemes without having a definite meaning and are not stand alone.

Lookup Table L

pronounList = ["ମୁଁ", "ତୁ", "ସବୁ", "ଆମେ", "ମୋ", "ତୁମ", "ଅନ୍ୟ", "ଏଠି", "ସେଠି", "ତାଙ୍କୁ", "ସେ", "ଆପଣ",...]]

Transliteration: ["mū", "tu", "sabu", "āme", "mo", "tuma", "anā", "eāhi", "seāhi", "tāku", "se", "āpā",...]

Gloss: ["me", "you", "all", "we", "mine", "your", "other", "here", "there", "to him", "he/she", "you",...]

puncList = [“⌂”, “,””, “;”, “:”, “\”, “\”, “?”, “*”, “(”, “)”, “{”, “}”, “[”, “]”, “-”, ...]

Transliteration: [“⌂”, “,””, “;”, “:”, “\”, “\”, “?”, “*”, “(”, “)”, “{”, “}”, “[”, “]”, “-”, ...]

Gloss: [Same as English, only full stop (.) is represented with a different symbol (⌂)]

postPosList = [“ଉପରେ”, “ତଳେ”, “ଆଗରେ”, “ପଛରେ”, “ଭିତରେ”, “ଭିତର”, “ବାହାରେ”, “ପୂର୍ବରୁ”, “ସଙ୍ଗେ”, “ଠାରୁ”, “ଛଡ଼ା”, “ରୁ”, “ଛାଡ଼ା”, “ରୁ”, “ମଝିରେ”, ...]

Transliteration: [“upare”, “ta⌂e”, “āgare”, “pachhare”, “bhitare”, “bhitara”, “bāhāre”, “purbaru”, “sa⌂ge”, “⌂hāru”, “chhar⌂ā”, “ru”, “duārā”, “ku”, “majhire”, ...]

Gloss: [“above”, “below”, “ahead”, “behind”, “inside”, “inside”, “outside”, “before”, “with”, “from”, “except”, “from”, “by”, “to”, “in the middle”, ...]

conjunctionList = [“ଓ”, “ନା”, “ଯେ”, “ଏବଂ”, “ଆଉ”, “ପୁଣି”, “ଆହୁରି”, “ମଧ୍ୟ”, “କିନ୍ତୁ”, “ମାତ୍ର”, “ଅଥଚ”, “ଅଥବା”, “ଦିଗ୍ଘା”, “ବା”,...]

Transliteration: [“o”, “nā”, “je”, “eba⌂”, “āu”, “pu⌂i”, “āhuri”, “madh⌂a”, “kintu”, “mātra”, “athacha”, “athabā”, “kimbā”, “bā”,...]

Gloss: [“and”, “no”, “that”, “and”, “and”, “again”, “more”, “also”, “but”, “only”, “yet”, “or”, “or”, “or”,...]

questionWordsList = [“କିଏ”, “କେ”, “କି”, “କ’ଣ”, “କଣ”, “କେମିତି”, “କାହିଁକି”, “କିପରି”, “କେଉଁଠି”, “କେତେବେଳେ”, “କେବେ”, “କେଉଁ”, “କେତେ”, ...]

Transliteration: [“kie”, “ke”, “ki”, “ka’⌂a”, “ka⌂a”, “kemitī”, “kāhīki”, “kipari”, “keū⌂hi”, “ketebe⌂e”, “kebe”, “keū”, “kete”,

Gloss: [“who”, “who”, “who”, “what”, “what”, “how”, “why”, “how”, “where”, “when”, “when”, “which”, “how much”, ...]

nounSuffixList = [“କୁ”, “ରୁ”, “ରୁ”, “ଠାରୁ”, “ଛାଡ଼ା”,...]

Transliteration:[ku”, “⌂ku”, “ru”, “⌂hāru”, “duārā”,...]

Gloss: [“to”, “to”, “from”, “from”, “by”,...]

verbSuffixList = [“କରି”, “ଉଅଛି”, “ୁଅଛି”, “ଛାଡ଼ି”, “ୁଛି”, ...]

Transliteration: [“kari”, “uachhi”, “uachhi”, “uchhi”, “uchhi”, ...]

Gloss: These morphemes or suffixes are not stand alone and do not have exact meaning. They have a grammatical function.

nounPrefixList = [“ପ୍ର”, “ପରା”, “ଅପ”, “ସମ୍”, “ନି”, “ଅଧ୍”, “ସ୍ୱ”, “ନି”, “ଉର୍”, “ପରି”, “ପ୍ରତି”, “ଅବ”, “ଅନ୍ତ”, “ରୁ”, ...]

Transliteration: [“pra”, “parā”, “apa”, “sam”, “ni”, “adhi”, “su”, “ni”, “ut”, “pari”, “prati”, “aba”, “anu”, “du”, ...]

adjectiveSuffixList = [“ଜନକ”, “ଦାୟକ”, “କର”, “ହାନ”, “ମୟ”, “ତମ”, ...]

Transliteration: [“janaka”, “dā᳚aka”, “kara”, “hina”, “ma᳚a”, “tama”, ...]

Gloss: These morphemes (noun prefixes and adjective suffixes) are not stand alone and have no exact meaning. They have a grammatical function.

prePositionList = [“ଶ୍ରୀ”, “ଶ୍ରୀମତି”, “ଶ୍ରୀଯୁକ୍ତ”, “ତଃ”]

Transliteration: [“sri”, “srimati”, “srijukta”, “᳚a᳚”, ...]

Gloss: [“Mr”, “Mrs”, “Mr”, “Dr”, ...]

particleList = [“ନାହିଁ”, “ହଁ”, “ନା”, “ଆଦି”, ...]

Transliteration: [“nāhī”, “hī”, “na”, “ādi”, ...]

Gloss: [“no”, “yes”, “no”, “etc”, ...]

Derived rule-set

(i) Suffix rules

• **Noun**

Some of the noun suffixes with their transliterated form are:

କୁ, କୁ, ରୁ etc.

(ku, ᳚ku, ru)

ରିତାକୁ ବହି ଦିଅ । (ritāku bahi dia .)

Translation: Give book to Rita.

Here (ritāku) belongs to noun category and can be identified by its “ku” suffix.

◦ **Verb**

Some of the verb suffixes are:

ଉଛି, ଉଥାନ୍ତି, ଉଛନ୍ତି, ଛନ୍ତି etc. (uchhi, uthānti, uchhanti etc)
e.g. ହୋଇଛି, ହୋଇଥାନ୍ତି, କରିଥିଲା, ପଡ଼ିଛନ୍ତି

(hoichhi, hoithānti, karithilā, paichhanti) are all words belonging to verb.
Gloss: Done, happens, done, fallen

◦ **Adjective**

Example of few adjective suffixes with their transliterated form in the bracket are:

ଜନକ(janaka), ଦାୟକ(dāyaka), କର(kara), ହିନ(hina), ମୟ(mayā), ତମ(tama), ାୟ(iya), ିୟ(iyā)
e.g. ଶୋଚନୀୟ, ଉଚ୍ଚତମ, ଫଳଦାୟକ

(sochaniā, uchatama, phaādāāka) are all words belonging to adjective. Gloss: Severe, highest, fruitful.

(ii) Prefix rules

Some of the noun prefixes are:

ଆ, ପରା, ପ୍ର (ā, parā, pra)

e.g. ଆମେରିକା, ପରାକ୍ରମ, ପ୍ରତିଷ୍ଠା (āmerikā, parākrama, pratisthā)

Gloss: America, courage, prestige

(iii) Starting word of a sentence is generally a noun or a pronoun in comparison to other tags.

e.g. ପୂଜା_NN ଇଛାପୁର_NN ଗ୍ରାମର_NN ଶିକ୍ଷିତ_JJ ଝିଅ_NN ।_PUNC

Transliteration: pujā_NN ichhāpura_NN grāmara_NN sikhita_JJ jhia_NN._PUNC

Translation : Puja is an educated girl of Ichapur village.

ତା_PR ଭଲିଆ_PSP ଭଲ_JJ ପିଲା_NN ମିଳିବା_NN କଷ୍ଟ_NN ।_PUNC

Transliteration: tā_PR bhaīā_PSP bhala_JJ pilā_NN miībā_NN kasā_NN._PUNC

Translation: To get a nice girl like her is difficult.

(iv) Starting word of a sentence is rarely an adjective.

(v) A quantifier is followed by a noun or adjective or an adverb not any other part of speech.

e.g. ବହୁତ QT ଲୋକ NN, ବହୁତ QT ଭଲ JJ ଲୋକ NN

(bahuta_QT loka_NN, bahuta_QT bhala_JJ loka_NN)

Translation: more people, more good people

(vi) Verb generally comes after object (noun).

(vii) The list of conjunction like

ଓ, ବା, କିନ୍ତୁ etc.

(o, bā, kintu) are few and can be maintained in the table.

(viii) The list of postpositions like

ଉପରୁ, ତଳୁ etc.

Translation: up, down

(uparu, taଁu etc.) is maintained in the Lookup table.

(ix) The list of punctuation like full stop (.), question mark (?), exclamation mark (!) etc. are few and is maintained in the Lookup table.

(x) If the succeeding word is postposition then the current word is a noun or a pronoun. If the current word is not in the pronoun list then it is tagged as noun.

(xi) If the preceding word is a pronoun, succeeding word is a noun and the current word is not in the pronoun list then the current word is an adjective.

(xii) If the current word is postposition and the succeeding word is tagged verb then change the current word to adverb.

Sagarika Pattnaik received her Bachelor in Engineering in Computer Sc and Engineering from Seemanta Engineering College (North Orissa University), Orissa, India in 2001. Received her Master's in Computer Sc. & Informatics from ITER (S'O'A Deemed to be University) Bhubaneswar, Orissa, India in 2013. Presently working as a lecturer in Computer Sc. in P.N. Autonomous College, Khordha, Orissa, India. Her research interests include natural Language processing, artificial intelligence, communication and network and wireless sensor network.

Ajit Kumar Nayak is a professor and HOD of the Department of Computer Science and Information Technology, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha. He graduated in Electrical Engineering from the Institution of Engineers, India in the year 1994, obtained M. Tech. and Ph. D. degree in Computer Science from Utkal University in 2001 and 2010 respectively. His research interests include Computer Networking, Ad Hoc & Sensor Networks, Machine Learning, Natural Language Computing, Speech and Image Processing etc. He has published about 55 research papers in various journals and conferences. Also co-authored a book 'Computer Network Simulation using NS2', CRC Press. He has also participated as an organizing member of several conferences and workshops in International and National level.