ODIA PARTS OF SPEECH TAGGING CORPORA: SUITABILITY OF STATISTICAL MODELS

Thesis submitted to

Jawaharlal Nehru University

in partial fulfilment of the requirements for the award of the degree of

MASTER OF PHILOSOPHY

Supervised by

DR. GIRISH NATH JHA

Submitted by

PITAMBAR BEHERA

Centre for Linguistics

School of Language, Literature and Culture Studies Jawaharlal Nehru University

New Delhi, India.

2015

Date: 22nd July, 2015

DECLARATION BY THE CANDIDATE

This thesis titled **"Odia Parts of Speech Tagging Corpora: Suitability of Statistical Models"** submitted by me for the award of the degree of **Master of Philosophy**, is an original work and has not been submitted so far in part or in full, for any other degree or diploma of any University or Institution.

Pitambar Behera

(Mr. Pitambar Behera)

M. Phil Student

Centre for Linguistics,

SLL & CS

JNU

Centre for Linguistics
School of Language, Literature & Culture Studies
Jawaharlal Nehru University
New Delhi-110067, India

Date: 22nd July, 2015

CERTIFICATE

This thesis titled **"Odia Parts of Speech Tagging Corpora: Suitability of Statistical Models"** submitted by **Mr. Pitambar Behera** to the Centre for Linguistics, School of Language, Literature and Culture Studies, Jawaharlal Nehru University, New Delhi, for the award of the degree of **Master of Philosophy**, is an original work and has not been submitted so far in part or in full, for any other degree or diploma of any University or Institution.

This may be placed before the examiners for evaluation for the award of the degree of Master of Philosophy.

Girishnath Jha

(DR. GIRISH NATH JHA)
SUPERVISOR

SUPERVISOR
Dr. GIRISH NATH JHA
Associate Professor
Special Centre for Sanskrit Studies
Jawaharlal Nehru University
New Delhi - 110067

(PROF. AYESHA KIDWAI)
CHAIRPERSON

Chairperson
Centre for Linguistics
School of Language, Literature & Culture Studies
Jawaharlal Nehru University, New Delhi-110067

Dedicated to

my dear

mother, grand father, father,

family members,

teachers

and

lovable friends

who have been really inspirational for me throughout my entire academic career...

Acknowledgements

Firstly, I extend my gratitude to Dr. Girish Nath Jha, my guide, supervisor, and director of the project entitled Indian Languages Corpora Initiative (ILCI). In the able hands of Dr. Jha, I have been inducted into the vast arena of computational linguistics (CL), Natural Language Processing (NLP) and Artificial Intelligence (AI). He has been instrumental in my induction into more specific fields such as parts of speech tagging, statistical modelling for Indian languages, computational morpho-syntax and syntax. We have learnt from him how to create statistical models understanding the structures of the languages. Without him this novel undertaking could never be accomplished.

I am really indebted to all my teachers at the Centre for Linguistics, School of Language, Literature and Culture Studies, Jawaharlal Nehru University from whom I have learnt the rudiments of linguistics. I owe my thankfulness to Prof. Ayesha Kidwai for my initiation into the fields of morpho-syntax and syntax. Although I have not been directly benefitted from her, her timely advice, researches in syntax, books, papers, and finally notices on the notice board made me stick to my research for its timely completion. I am especially indebted to former Prof. Anvita Abbi who has initiated me into the field linguistics, endangered and lesser-known languages. From her teachings, I have learnt a great many things regarding less-resourced, endangered, and less-described languages. I am thankful to Prof. P. K. S. Pandey who taught and guided me into the area of research methodology. Thanks are also due to Prof. Vaishna Narang who initiated us to the area of applied linguistics and research methodology. I acknowledge Prof. Franson Manjali, because it is he who has developed in us a philosophical bent of mind and making us capable in conducting research. I also thank Prof. Pradeep Kumar Das who has introduced us the structures of the lesser-known languages; for always being approachable, and for his kind suggestions for improvement.

I acknowledge the assistance from the Director of the ILCI Project {under TDIL (Technology Development for Indian Languages) funded by the Department of Information and Technology, Government of India} for providing me half of the Odia annotated corpora. Further, I acknowledge the parts of speech annotation works under ILCI project conducted by Professor Panchanan Mohanty, Centre for Applied Linguistics and Translation Studies, University of Hyderabad and Dr. Bijaylaxmi Das,

Department of Odia, Ravenshaw University, Cuttack, Odisha. Without their great endeavour in parts of speech annotation, this could not be completed.

Special thanks are also due to Atul Kumar Ojha who has been always ready to assist me at any time linguistically and computationally. It is he who has configured both the statistical models employed in the concerned research.

I thank Sharmin Muzaffar and Dhruwa Shankar Prasad for a fine proof-reading of the nontechnical aspects of the texts in this dissertation. I also thank Abhishek Kumar Sharma and Abhishek Mishra for helping me write a java tokenizer programme and a program for precision and recall calculation during CRF++ evaluation. I also thank Abhishek Kumar Jha, CL Lab Assitant, SCSS, for eradicating problems in the computer during my research.

Above all, a million thanks to the Computational Linguistics Research and Development Lab in the Special Centre for Sanskrit Studies for providing me access to use computers during the designing of the tool **“Odia Parts of Speech Taggers”**.

-Pitambar Behera

CONTENTS

CHAPTER 4

List of Abbreviations Used

BIS: Bureau of Indian Standardization

BNC: British National Corpus

CDAC: Centre for Development of Advanced Computing

CGC: Computational Grammar Coder

CIEFL: Central Institute of English and Foreign Languages

CIIL: Central Institute of Indian Languages

CLAWS: Constituent Likelihood Automatic Word Tagging System

COCA: Corpus of Contemporary American English

CRF++: Conditional Random Fields

DeitY: Department of Electronics and Information Technology

EFLU: The English and Foreign Languages University

HMM: Hidden Markov Model

IA: India Languages

IIIT: International Institute of Information Technology

ILCI: Indian Languages Corpora Initiative

ILMT: Indian Language Machine Translation

ISI: Indian Statistical Institute

JJP: Adjectival Phrase

LDC-IL: Linguistic Data Consortium for Indian Languages

LTRC: Language Technologies Research Centre

MA: Morphological Analyser

MBT: Memory-Based Tagger Generation and Tagging

ME: Maximum Entropy

MIA: Middle Indo-Aryan

MSRI: Microsoft Research India Private Limited

MT: Machine Translator

NER: Named Entity Recognition

NIA: New Indo-Aryan

NLP: Natural Language Processing

NP: Noun Phrase

OIA: Old Indo-Aryan

OOA: Object Oriented Approach

PNG: Person Number Gender

POS: Parts of Speech

RBP: Adverbial Phrase

SP: Shallow Parser

SVM: Support vector Machine

TAM: Tense Aspect Mood

TL: Target Language

TMC: The Time Magazine Corpus

TDIL: Technology Development for Indian Languages

TTS: Text To Speech

UCRL: University Centre for Corpus Research

VP: Verb Phrase

WSJ: Wall Street Journal

List of BIS Abbreviations Parts of Speech Tags

CC_CCD: Coordinating Conjunction

CC_CCS: Subordinating Conjunction

DM_DMD: Deictic Demonstrative

DM_DMI: Indefinite Demonstrative

DM_DMQ: Interrogative/wh Demonstrative

DM_DMR: Relative Demonstrative

JJ: Adjective

N_NN: Common Noun

N_NNP: Proper Noun

N_NST: Spatial and Temporal Noun

N_NNV: Verbal Noun

PSP: Postposition

PR_PRC: Reciprocal Pronoun

PR_PRF: Reflexive Pronoun

PR_PRI: Indefinite Pronoun

PR_PRL: Relative Pronoun

PR_PRP: Personal Pronoun

PR_PRQ: Interrogative/wh Pronoun

QT_QTC: Cardinal Quantifier

QT_QTF: General Quantifier

QT_QTO: Ordinal Quantifier

RB: Adverb

RD_ECH: Reduplicative Echo Word

RD_PUNC: Punctuation

RD_RDF: Foreign Word

RD_UNK: Unknown Word

RD_SYM: Symbol/Special Character

RP_CL: Classifier

RP_NEG: Negation

RP_INJ: Interjection

RP_INTF: Intensifier

RP_RPD: Default Particle

V_VAUX: Auxiliary Verb

V_VM: Main Verb

V_VM_VF: Finite Verb

V_VM_VINF: Infinitive Verb

V_VM_VNF: Non-Finite Verb

V_VM_VNG: Gerundive Verb

List of Figures, Screenshots, Charts, Tables, Snippets and Maps

CHAPTER 1

1. INTRODUCTION

This introductory chapter is divided into five major sections. The very initial section provides an introduction to the Odia language, the geographical distribution of the state of Odisha, prominent languages spoken in the state, the evolution of the language historically and the script used for writing. The second section contains the review of linguistics and computational linguistics literature. The third section encapsulates different types of parts of speech annotation, its application, aims and objectives of the undertaken research, hypothesis, research questions, and rationale for the study and the uniqueness of the current research. The fourth section introduces the computational task to be dealt with, a precise description of the problem and solution. Further, it includes a brief introduction to the statistical models namely, SVM and CRF++ and a short comparison. The final section of the chapter which is research methodology deals with methods of data collection, annotation and analysis.

1.1.The Odia Language

Odia[1] /ɒɖɪɑ/ is recently declared as one of the classical languages (Pattanayak and Prushty, 2013; Jha et al., 2014) of India including already existing five; earlier, it was a Scheduled Language under the Eighth Schedule of the Constitution of India. It owes its genesis to the Indo-Aryan language family and was formerly known as Oriya. "Odisha is the modern name of the ancient Kalinga Empire, variously known as Udra, Utkala, Kalinga, Kosala, Toshala and Kangoda in different periods in history" (Pattanayak and Prushty, 2013). Apart from the fact that it inherits most of the salient linguistic features from the Indo-Aryan (IA) group, it also has some features pertaining to the Dravidian languages; as it spreads in an adjacent area where both the IA and Dravidian languages converge (Pattanaik, 2004). The Odia speaking population amounts to 41, 974, 24218[2] as reported by the Population Census, the Government of India 2011. It is also spoken in the neighbouring states[3] of Odisha (formerly Orissa) (see Fig. 1.), some parts of West

[1] The nomenclature 'Oriya' has been formerly used which has been changed to 'Odia' and similarly the state of 'Orissa' changed to 'Odisha'. So, in the dissertation, 'Odia' has been used.
[2] http://www.censusindia.gov.in/2011documents/lsi/ling_Orissa.html
[3] www.ethnologue.com

Bengal, Chattisgarh, Jharkhand, Andhra Pradesh and by the overseas population in U.S. and U.K. and in some other countries. Furthermore, it is the language of the government, literature, used as the medium of instruction in pedagogy and the so-called the standard form of the language. In addition, the attitude of the speakers towards the language is quite positive and the language is vital.[4]

As early as the 1870s one of the western scholars John Beams has stated that by the time Odia became "a fixed and settled language", its 'sister languages' like Bengali and Assamese even did not exist. The Bengalis used to speak a "variety of corrupt forms of Eastern Hindi" at that time (Pattanayak and Prushty, 2013).

As cited in Pattanayak and Prushty (2013), Suniti Kumar Chatterjee observes that out of the three languages (Bengali, Odia and Assamese), Odia is the 'eldest sister language' and has been able 'to preserve its archaic nature' with regard to grammar and pronunciation. One of the reasons of its split could be that it may have "branched off from the parent language" prior to the separation of the Bengali and Assamese languages from the Eastern group of IA languages.

Like most of the Indian languages, Odia is also a resource-poor language with less computerization. Some of the NLP tools that have been developed are either not available online or have not been made public or half-finished. For resource-rich languages like English there are many electronically annotated available corpora of huge volume, for instance, Wall Street Journal (WSJ)[5] corpus, Corpus of Contemporary American English (COCA)[6], British National Corpus (BNC)[7], The Time Magazine Corpus (TMC)[8] and so on. Considering the NLP situation in Indian languages, they are poor so far as the availability of electronically annotated written corpora is concerned. One of the successful attempts has been to build a written corpus of around 100k sentences under the ILCI Project[9] (Banerjee et al., 2013; Jha, 2010) in both the phases, funded by the Department of Information Technology (DIT), Govt. of India.

[4] www.ethnologue.com

[5] https://catalog.ldc.upenn.edu/LDC2000T43

[6] http://corpus.byu.edu/coca/

[7] http://www.natcorp.ox.ac.uk/

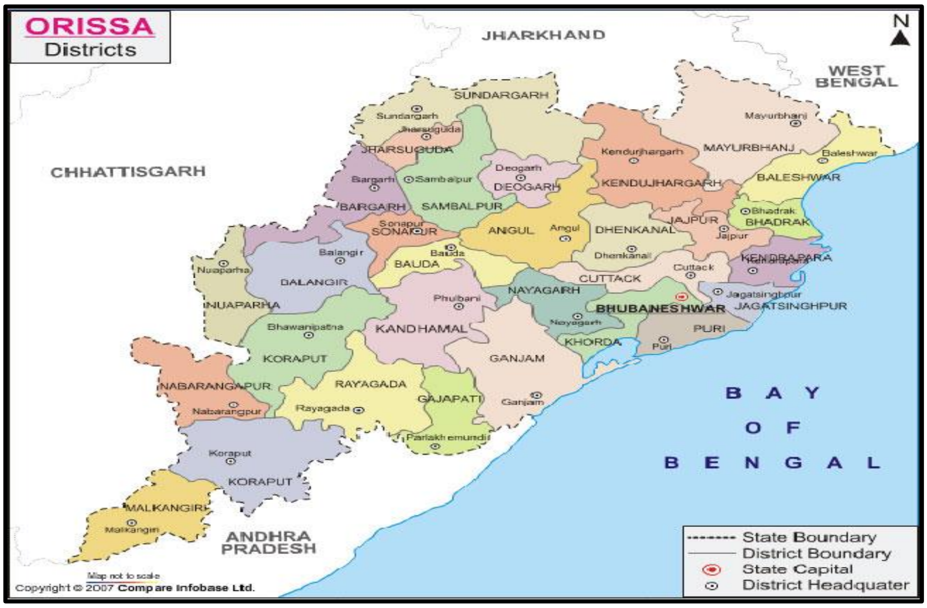
[8] http://corpus.byu.edu/time/

[9] http://ildc.in/Oriya/Oindex.aspx

Some other attempts have also been initiated by Language Technologies Research Centre (LTRC), International Institute of Information Technology (IIIT) Hyderabad,[10] Centre for Development of Advanced Computing (CDAC),[11] spoken corpus by Central Institute of Indian Languages (CIIL), Emille Corpus by Lancaster University, UK and so on. Thus, most of the corpora that have been collected are speech corpora except the CIIL, ILCI and IIIT Hyderabad.

1.1.1. Geographical Distribution

The territory of Odisha ranges from latitude 17 degree 31' to 20 degrees 31' N and its longitude extends from 81 degrees 31' to 87 degrees 30' (SER Orissa, 2007). Odisha is one among the twenty-nine states in the Federal Union of India surrounded by the Bay of Bengal in the east, W.B. in the north-east direction, the state of Jharkhand in the north side, A.P. in the southern part, and Chattisgarh in the west. It has an area of approximately 1, 55, 7072 k.m. with a vast coastline of around 480 k.m. As far As the morphological texture of the whole landmass of Odisha is concerned, it can be categorized into five major parts: the eastern coastal plains, 'the middle mountainous and highlands' region in the north and northwest parts, the plateaus of the central part, the rolling uplands of the western parts and 'the major flood plains'.[12]



Map. 1. The Political Map of Odisha Adapted from the Linguistic Survey of India

[10] http://ltrc.iiit.ac.in/showfile.php?filename=ltrc/internal/nlp/corpus/index.html
[11] http://www.cdac.in/index.aspx?id=mc_ilf_indian_language_fcdt
[12] http://www.orissatourism.org/orissa-geography.html

1.1.2. Prominent Languages of Odisha

There is no large-scale difference between the dialects and language in the present-day sociolinguistics and talking about these issues is considered to be a cliché. But, some of these issues are still present in a subtle manner at various levels in the society. These issues pop up, when a particular state in a federation demands a separate state on the basis of language. Therefore, in spite of it being considered as a cliché, the dialects and languages of Odia have been discussed on the basis of the above-said rationale using the traditional terminologies.

The alternate nomenclatures for Odia are "Odisha, Odri, Odrum, Oliya, Uriya, Utkali, Vadiya and Yudhia". The dialects of Odia language are Halbi, Midnapore Odia, Mughalbandi ('Odia Proper, Standard Odia'), North Balasore Odia, Northwestern Odia, Southern Odia, Western Odia (Sambalpuri).[13] Odia has approximately 75 to 76 percent of morphological similarity with Sambalpuri[14] dialect which is also considered to be a separate language by its native speakers. Odia further shares some common linguistic features pertaining to morphology, syntax and semantics with its prominent sister languages like Assamese and Bengali. The Eastern New Indo-Aryan (NIA) languages are the descendants of the 'Magadhan Apabhramsha' of around the seventh century A.D., which owes its origin to the 'Magadhan Prakrit' and probably in the language of the Ashokan inscriptions found in Dhauligiri rock edict in Orissa.[15] There are several languages and dialects of Odia which are spoken in different portions of Bengal, Bihar, Assam and Orissa. Some of the major dialects are Sambalpuri, Baleswari, Utkali, and Ganjami or Berhampuri. Sambalpuri is spoken by a large number of people in the ten districts of Western Odisha, parts of Jharkhand, West Bengal, and Chattisgarh. Bhojpuri, Maithili, and Magahi are largely spoken in Bihar, Uttar Pradesh and Jharkhand.

Broadly speaking, Odia has three prominent group of dialects namely, coastal dialect, western dialect and South Western dialects. The coastal variety is considered to be the 'modern standard language' and it has much affinity with the Bengali. The western group of dialect is spoken in the western part of Odisha and it has a fair amount of linguistic similarity with Kosli and Chattisgarhi. The South Western group of dialects

[13] www.ethnologue.com

[14] Note: there is a language-dialect issue with Odia

[15] For details about the linguistic history of Odia, see Majumdar 1970, Tripathi 1962 and Misra 1975

(Desiya, Bhatri, Jharin) has much resemblance with Halbi. This fact provides strong evidence of the Aryan language getting systematically expanded (Samantaray, 2008). Debajit Deb (2012) has stated that there are seven dialects of Odia: 'Midnapori Odia', 'Singhbhumi Odia', 'Baleswar/Baleswari Odia', 'Ganjami Odia', 'Desiya Odia', 'Sambalpuri Odia' and 'Bhatri'.

Masica (1991) has opined that the delta of the Mahanadi, flowing through the state and situated right at the southwest of Bengal, is 'the center of the Odia language'. Non-Aryan-speaking tribal people have been living in the state of Orissa and "a large block of which separate Odia from Bengali". The remote Sambalpur lowland part of Odisha 'has a distinctive dialect'. Bhatri is one of the 'aberrant dialects' of Odia initially spoken by the former 'Gond (Dravidian) tribesmen' in the northeast part of the pre-independent Bastar State; which is presently known by this nomenclature as one of the districts of Madhya Pradesh.

As cited in Pattanayak and Prushty (2013), Grierson has opined that "Odia is remarkably free from dialectic variations". But this statement is quite vague considering the present-day situation in Odisha. The well-known saying which upholds the fact and is true for all over the north of India that "language changes in 10 kilometers" also proves to be true in Orissa notwithstanding the fact that Grierson puts it in a wrong way. He provided the instance of a language namely the Mughalbundi, which consists of Cuttack, Puri and the southern half of Balasore and upholds that the language 'is one and the same'. This case cannot be generalized for every language or dialect for that matter. Therefore, it can be undoubtedly said that there are a large number of languages and dialects existing presently in Odisha.

1.1.3. Historical Development of the Language

According to Samantaray (2008), Odia can be classified as "part of the Magadhan Subgroup of the Indo-German Group of language" like Bengali and Assamese, its sister languages. Scholars uphold the idea regarding the origin of these languages that they must have originated from a common genesis at some point in history. This perspective is evident of the fact that a collection of Buddhist poems has been discovered from Nepal State Library by Sri Haraprasad Sastri entitled *Boudha Gaan O'Dohan.* Prof. Oldenburg, the German linguist, has averred that Pali could be the original or source language of Odisha. His decision was completely based on the Hati Gumpha inscription which was inscribed during the Kharavela's regime written in Pali.[16]

The Brahmi Indic scripts have a vantage place in the study of graphology. They are 'alpha-syllabic scripts' (Bright, 1996), which denotes to the fact that they are basically 'segmental in nature' because most of the segments are represented in the script. The Odia script (Mohapatra, 1996) derived from the Brahmi Indic script has its origin from the northern group of South Asian scripts.

According to L.S.S. O' Malley as cited in Pattanayak and Prushty (2013), "Odia is in an older stage of grammatical development than even classical Sanskrit, and, among Indo-Aryan Languages, can only be compared with the ancient Sanskrit spoken in the Vedic times". Dr. G.N. Das (2006) has vividly discussed the origin of the Odia language up to 1500 A.D. in his *History of Odia language.* Furthermore, he has provided a comprehensive overview about how the language has evolved from the OIA to the NIA. The developmental history of the Indo-Aryan languages can be divided into three stages: Old Indo-Aryan (OIA), Middle Indo-Aryan (MIA) and New Indo-Aryan (NIA) (see Fig.1). The development of some of these individual languages of these sub-groups has been rigorously studied from the linguistic point of view and the scholars have identified several of their linguistic features. In addition, Prof. Das has stated that the account of the literary developmental period especially of prose and poetry in Odia language, written in inscriptions, copper plate grants, and palm leaves, is a fascinating one.

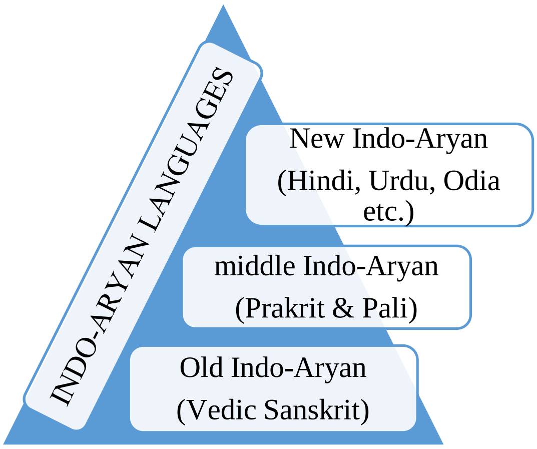

INDO-ARYAN LANGUAGES
New Indo-Aryan (Hindi, Urdu, Odia etc.)
middle Indo-Aryan (Prakrit & Pali)
Old Indo-Aryan (Vedic Sanskrit)

Fig. 1. The Development of the NIA Language Family

[16] See Pattanayak and Prushty, 2013 and Bright, 1996 for more information

1.1.4. The Script

As referred in Pattanayak and Prushty (2013), Odisha “is the only Indian state, where three types of Brahmi script have been discovered like- Pre-Brahmi, Brahmi & Post Brahmi”. The Indian script ‘o’, which was discovered from Yogimata rock painting of Nuapada district, is the primitive form of Indian script and is the first glimpse of possible origin of the Odia language and script. In relation to the paintings discovered at the Vikramkhol, K.P Jayaswal has stated, “the Vikramkhol inscription supplies a link between the passages of the letter from the Mohenjo-Daro script to Brahmi”.

The Odia script is a descendant of the Brahmi script[17] and is related to the other North-Indian scripts, such as Devanagari. The Odia (Mahapatra, 1996) script, derived from the Brahmi Indic script, has its genesis from the Northern group of South Asian scripts. As cited by Routray (2009) its origin can be traced back to the earliest disseminative alphabet of India, known as the Brahmi[18] of the third century B.C. discovered at Dhauli[19] “on the southern bank of the river Daya near Bhubaneswar and the other at Jaugada”[20] on the Rishikulya river bank in the district of Ganjam.

The cursive shape of the letters seems to have originated from the Southern script; since the Southern scripts are all written in cursive shape. In addition, it is traditionally an assumed notion that probably the necessity to write on palm leaves with pointed styles had motivated the cursive nature of the letter-writing system. The fundamental design of the Odia script is analogous to that of Devanagari. The basic design is that vowels occurring at the syllable initial place are represented with their own symbols. Otherwise, dependent vowel symbols are utilized. Hence, the orthographic syllable parsing procedure for Devanagari also functions well for Odia. The Brahmi Indic scripts are alpha-syllabic scripts (Bright, 1996a), which denotes the fact that they are basically segmental in nature as almost all segments are represented in the script. Therefore, it can be stated that Odia is graphologically represented as it is spoken.

As referred in Pattanayak and Prushty (2013), taking into account the accent in the Hatigumpha inscriptions, the French scholar S. Sylvan Levi has confirmed it of being written in the Pali in the Post-Brahmi script. The internationally-acclaimed German

[17] for the historical evolution, see Tripathi, 1962

[18] The earliest Indian script running from the left to the right is known as Brahmi script.

[19] CII, Vol. I, pp. 84 - 97 ff and plates. Orissa Review

[20] Ibid. Vol. l, pp, 101- 115 ff. and plates. Orissa Review

linguist, Professor Herman Oldenburg has mentioned that Pali was the original language of Orissa. According to the view of John Boulton (2003), the development of the Odia script owes to the fusion of Pali with the components from the 'aboriginal and Dravidian languages' that were used by the ancient inhabitants of Odisha and West Bengal.

1.2. Review of Linguistics and Computational Linguistics Literature in Odia

In this section, some of the research works in linguistics have been reported, but the computational research in Odia has been vividly reviewed.

1.2.1. Linguistics Research in Odia

As reviewed by Neukom (2003), The existing Odia grammar books available presently are not of huge volume, either obsolete, originating from the 19th century A.D. (e.g. Maltby's Odia Grammar in 1986), or hardly available (Matson's Odia Grammar in the year 1971), or accessible for Russian readers only (Karpushkin's Odia Grammar written in Russian in the year 1964). Some courses pertaining to language teaching have been developed by (Das Gupta 1980; Mohanty 1989; Pattanayak and Das 1972). But the point is they do not encapsulate a comprehensive analysis of all the grammatical categories. The other research-oriented works have been conducted focusing only on parts of the grammar: morphology, syntax, and historical evolution. (Bhattacharya 1993; Dash 1982; Misra 1975) have worked on the morphology of Odia while (Majumdar 1970; Pattanayak 1966; Tripathi 1962) have worked on the historical evolution of the language. So far as syntax is concerned, scholars like (anaphora: Patnaik, 1994; complementation: Bal, 1990; Patnaik, 1976; nonfinite clauses: Nayak, 1987; nominal phrases: Sahoo, 1996) have investigated.

Odia Inflectional morphology with the title *Descriptive Odia Morphology in the Paninian Model* has been investigated following the Paninian model by Tulasi Das Majhi in the year 2007, JNU. Besides, in the English and Foreign Languages University (E.F.L.U), formerly C.I.E.F.L, there have been many research works conducted in the fields of morphology and syntax.

1.2.2. Computational Research and Development in Odia

So far as the field of computational research in Odia language is concerned, it is not so computerized in comparison to other Indian languages. Sarala Font[21] for Odia has been developed. Odia Open-Type Unicode Jagannath Font has been designed by Sujata S. Patel in 1995.[22] Some of the freely available Odia fonts are Akruti,[23] Lohit, Samyak, ORB-TT Mukta EN Normal font etc. There are some other fonts as well; such as Aprant, Mahanadi, Brahmani, Tara, Khandadhar etc.[24] Fonts have also been developed by CDAC (Centre for Development of Advanced Computing and TDIL (Technology Development for Indian Languages) under the Department of Electronics and Information Technology (DeitY).

1.2.2.1.Corpora and Tagsets

Corpus collection is one of most prominent tasks and a stepping stone for research and developments in the field of natural language Processing (NLP). As defined by Crystal (1992), "Linguistic corpus is a collection of data, either in written form or in the form of recorded speech". CIIL, ILCI[25] Project under TDIL, ILMT Project under IIIT Hyderabad have collected corpora in Odia. CIIL has developed annotated cultural vocabulary in Odia under the leadership of the Utkal University of Culture.[26] ILCI Project has created around 50k multilingual translated corpora in the first phase with Hindi as the source language while in the second phase other 50k sentences have been parts of speech annotated and chunking is under process. IIIT Hyderabad has created a corpus for around 12 Indian languages. Under the Indian Languages Corpora Initiative (ILCI) project initiated by the DeitY, Govt. of India, Jawaharlal Nehru University, New Delhi have collected corpus in Hindi as source language and translated it into Odia as target language along with other scheduled Indian languages incorporated in the project in the first phase. Presently, the project is in the second phase where another 50k sentences have been collected for parts of speech annotation and chunking. Odia annotated speech corpora has been developed by LDC-IL (Linguistic Data Consortium for Indian Languages), CIIL.[27] "This Odia Speech Recognition database was collected

[21] http://oriya.indiatyping.com/index.php/download-oriya-font
[22] http://www.odialanguage.com/Odia_fonts.html
[23] http://www.aparts.org/products/aprant-font-odia-key-board-manager/odia-fonts/
[24] http://www.aparts.org/products/aprant-font-odia-key-board-manager/odia-fonts/
[25] http://sanskrit.jnu.ac.in/projects/ilci.jsp?proj=ilci
[26] http://www.ciil.org/ProgReportworkshop3.aspx
[27] http://www.ldcil.org/resourcesSpeechCorpOriya.aspx

in Orissa and contains the voices of 450 different native speakers who were selected according to age distribution (16-20,21-50,51+), gender, dialectical regions and environment (home, office, and public place)".[28] The written corpora collected by CIIL is 1, 521, 181.[29] Indian Institute of Applied Sciences, Bhubaneswar developed corpus for Odia. B. R. Das, S. Patnaik, and N. S. Dash have developed corpus for Odia from the domain of newspapers. S. S. Nanda, S. Mishra, S. Mohanty have created text mining platform for tourists. Pitambar Behera has collected and documented corpora on Imagact[30] and TypeCraft[31] platforms. The tagsets for Odia have been developed by CIIL Mysore (LDC-IL), Microsoft Research, India (IL-POSTS), BIS (DeitY), IIIT Hyderabad (ILMT).

1.2.2.2.NLP Tools

CDAC and ISI (Indian Statistical Institute) Kolkata have developed an OCR for Odia. B. R. Das and S. Patnaik have made a Single Neural Network POS Tagger using the artificial neural network. Rajkumar has researched on POS Tagging in Odia. S. Mohanty, P. Ku. Santi, and K.P. Das Adhikary has made a morph analyzer testing with OriNet. K. R. Shatabdi has presented a *Finite-State Morphological Processor of Odia verbal Forms*. R. Ch. Balabantray and S. Ku. Lenka have created a Computational Model for the Reduplication feature of Odia. D. P. Sethi has developed a morph analyzer for Odia. A Multiword Chunker has been developed using lexical knowledge base by M. Ku. Jena, S. Mohanty, and R. Balabantaray. Odia Language Shallow Parser applying machine learning approach has been developed by M. Ku. Jena and R. Balabantaray. Mr. Vijayanand Kommaluri, and Mr. Subramanian Ramalingam from Pondicherry University. Shallow Parser Tools for Indian Languages under which the parser for Odia has been developed by Dr. Panchanan Mohanty, UOH, Hyderabad. A Support Vector Machine Named Entity Recognizer for Odia Language was developed by B.R. Das and S. Patnaik. Named Entity Recognizer has been developed by D. Swain and C. Pati. *A Hybrid Odia Named Entity Recognition system: Harnessing the Power of Rule* was developed by S. Biswas, S. P. Mishra, S. Acharya, & S. Mohanty. Odia WordNet[32] has been developed by Prof. Panchanan Mohanty, UOH, Hyderabad.

[28] http://www.ldcil.org/resourcesSpeechCorpOriya.aspx
[29] http://www.ldcil.org/resourcesTextCorp.aspx
[30] http://www.imagact.it/imagact/query/dictionary.seam
[31] http://typecraft.org/tc2wiki/Main_Page
[32] http://indradhanush.unigoa.ac.in/odiawordnet

English to Odia online dictionary and for Android has been developed. Google Odia Translator Project is currently undergoing under the guidance of Dr. Sashi Bhusan Maharana. Utkal University, Bhubaneswar has developed AnglaOdia MT under the AnglaBharti Mission.

1.2.2.2.1. Morphological Analyzers

As reviewed by Sethi (2014) in *A Survey on Odia Computational Morphology*, Itisree Jena, Sriram Chaudhury, Himani Chaudhry, and Dipti M. Sharma presented a paper entitled *Developing Odia Morphological Analyzer Using Lt-toolbox*. The analyzer for Odia developed by them employs 'the paradigm approach' which is a method that defines all the word forms of a given stem and the structure of their features. The system is responsible for tackling only the inflectional morphology of the grammatical categories such as noun, verb, and adjectives.

R. C. Balabantray, M. K. Jena, and S. Mohanty presented a paper entitled *Shallow Morphology based complex predicates extraction in Odia*. The aim of the paper was to extract the complex predicates for the Odia sentences that contained

"The lexicon pattern {[MMM] (n/adj) [NNN] (v)} in the shallow parsed sentence where MMM and NNN represent any word. The lexical category of the root word of MMM is either noun (n) or adjective (adj) and the lexical category of the root word is a verb (v)".[33]

Sanghamitra Mohanty, Prabhat Kumar Santi, and K. P. Das Adhikary have developed a Morph Analyzer testing with OriNet. They designed the architecture of Odia Morphological Analyzer (OMA) which comprised five parts e.g. OriNet database (OD), OMA Engine (OE), Morphological Parser (MP), and Decision Tree (DT). The OD is responsible for storing the lexicon of Odia language, OE 'processes the system' while MP parses the word morphologically as according to the rules encoded. The function of the DT is to categorize all the morphemes of a given input word by drawing trees. They have further stated in the research paper that their application has been designed on the basis of 'object oriented approach' (OOA).

Kalyani R. Shabadi presented *Finite-State Morphological Processing of Odia Verbal Forms*. In this paper, she discusses the morphological processing of the verbal

[33] Ibid.

forms in Odia in a 'deterministic finite state automation'. This work proposes a computational model for designing an architecture for a morphological analyzer of Odia verbal forms "which can provide lexical, morphological, and syntactic information for each lexical unit in the analyzed verbal forms".

Balabantray and Lenka presented a paper with the title *Computational Model for Reduplication in Odia*. In this paper, they have examined the internal structure of Odia reduplication and an infinite number of possible generation of reduplicative words from a finite number of lexical categories.

Sethi (2013) has developed a Morphological Analyzer for Sambalpuri Inflected Verbal Forms. He has presented the morphological analyzer of a dialectal language employing the suffix stripping algorithm to develop the tool.

1.2.2.2.2. Parts of Speech Tagger

Das and Pattnaik (2014) have developed a Single Neural Network-based POS tagger for Odia language. Initially, the tagger has been selected empirically 'with a definite length of contextual information'. After that, multiple neurons comprising of a number of single neurons have been presented of a definite number. But they consist of a different length of contexts. The statistical tagger annotates the input data based on the voting on the output of all single-neuron tagger. It provides eighty one percent of accuracy.

1.2.3. Literature Review of Parts of Speech Tagging Research

This section has been divided into two major sub-sections: POS tagging in English and Indian languages.

1.2.3.1.Parts of Speech Tagging in English and Indian Languages

It was quite difficult to review all the literatures in parts of speech tagging in English and Indian languages. So, to the best of knowledge and availability of related data, the review has been conducted.

1.2.3.1.1. POS Tagging in English

The first system which is recorded is the UPENN in the years 1958-59. In the year 1963, Klein and Simmons constructed a Computational Grammar Coder (CGC). In 1971, Greene and Rubin developed a tagger TAGGIT which correctly tagged 77% of the Brown Corpus (American English Corpus) (Jurafsky and Martin, 2002, pp. 318). In the late 1970s, LOB (Lancaster-Oslo-Bergen) corpus used a tagger called CLAWS1[34] with probabilistic algorithm.

Recent stochastic algorithms use various statistical and machine-learning tools for estimating the probability of a given tag of a particular token. These algorithms apply a large amount of information like the context of the word, what their POS categories are, and the orthographic, and morphological features as well. Some of the noteworthy taggers are mentioned below.

Trigrams ‘n’ Tags (TNT)[35] is a stochastic HMM tagger based on ‘trigram analysis’ which uses a suffix analysis technique based on properties of words like suffixes in the data of training corpus, to evaluate lexical probabilities for unknown words having same suffixes. The tagger implements the Viterbi algorithm, is adaptable to any language and possibly any tagset. Tree Tagger[36] was developed by Helmud Schmid at the CL of the University of Stuttgart (1993-1996). It has been successfully applied to most of the European languages, Chinese, and some Slovenian languages. It uses HMM model and decision tree for smoothing and is adaptable to other new languages.

Stanford Log-linear Part-Of-Speech Tagger[37] is an open source software and a model for English, Arabic, Chinese, and German. This tagger is based on the Maximum Entropy framework. It can be trained on any language on a POS-annotated training text for the language. It was originally developed by Kristina Toutanova. Later, scholars like Manning, Klein, Morgan, Rafferty, and others improved on its reliability, efficiency, and usability.

Eric Brill introduced a POS tagger in 1992 that was rule-based called as Brill’s rule-based pos tagger. In this tagger, the grammar is induced directly in the form of handwritten linguistic rules to the training corpus and the performance is measured. He observed that the rule-based tagger is at par with the stochastic tagger in terms of quality, reliability, and efficiency.

[34] http://www.comp.lancs.ac.uk/ucrel/claws/trial.html

[35] http://www.coli.uni-saarland.de/~thorsten/tnt/

[36] http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

[37] http://nlp.stanford.edu/software/tagger.shtml

Chris Biemann's UNSUPOS– unsupervised POS tagging[38] is an open source tagger. It does not require an annotated training corpus. Instead, word categories are determined by analyzing a large sample of monolingual, sentence-separated plain text. The corpus it requires is really cumbersome. It requires approximately 100k sentences or two million tokens to start with. With the increasing data, it starts providing better performances.

MBT[39] (Memory-Based Tagger Generation and Tagging) "is a memory-based tagger-generator and a tagger". The former part can generate a sequence tagger "on the basis of a training set of tagged sequences" while the latter part can annotate new sequences based on the acquired memory during the training period. It has already been successfully applied to other NLP tasks like NER, IE, and disfluency in transcribed text.

CLAWS[40] (Constituent Likelihood Automatic Word Tagging System) is developed by UCREL and is based on the word ending and then uses an HMM method for estimating the most likely word and tag in each context. It was applied to annotate hundred million words of the BNC. The tagger has achieved 96 to 97 percent accuracy rate. It has 3.3 percent ambiguity rate and 1.5% error rate considering major POS categories.

1.2.3.1.2. Parts of Speech Tagging in Indian Languages

The POS tagger for Odia as per standard POS conventions has not been completed till date, except the Neural Network technique and an SVM without having a good POS guideline. Some of the POS research works for Indian languages have been summarized below.

- **Odia Neural Network Tagger:**

Das and Pattnaik (2014) have proposed a Single Neural Network-based parts of speech tagger for Odia language. The tagger has been selected empirically with the fixed length of context initially. Thereafter, multiple neurons, comprising of a number of single neurons, have been presented of fixed number but of different length of contexts. The tagger annotates the input data based on the voting on the output of all single-

[38] http://wortschatz.uni-leipzig.de/~cbiemann/software/unsupos.html
[39] http://ilk.uvt.nl/mbt/
[40] http://ucrel.lancs.ac.uk/claws/

neuron taggers. All the errors have been corrected through ‘forward propagation’ and then the corrected values of neurons have been transferred by the ‘feed-forward technique’ existing in the multiple layers. The multiple layers are the input, output, and the middle layers or the hidden layers. Owing to the fact that HMM-based technique does not produce competitive accuracy, a morphological analyzer has been applied for increasing the efficiency of the tagger. As has been reported, the tagger has an accuracy rate of 81 percent.

- **Odia SVM Tagger:**

Das et al., (2015) have developed an SVM Tagger with a training set of 10k and have reported an accuracy of 82%, a slight one percent increase in the accuracy rate of the earlier Neural Network tagger. The tagset used by them consists of only five tags and they have taken care of the features of different pos categories along with careful handling of prefixes and suffixes. For increasing the accuracy rate of the pos tagger, a set of lexicon consisting of around 200 words has been used.

- **Rule-based POS Tagger for Sanskrit Language:**

R. Chandrashekhar[41] (2002-2007) developed a rule-based tagger for Sanskrit language as a part of his doctoral research work. A stable POS tagset for Sanskrit text has a total of 134 tags: which has 65 number of word-level tags, 43 feature sub-tags, 25 punctuation tags, and one tag UN to tag unknown words.[42]

- **Stochastic Tagger for Sanskrit:**

Oliver Hellwig is the progenitor of the Sanskrit stochastic Tagger, which is a tagger for un-pre-processed Sanskrit text. The tagger employs a Markov model for tokenization and performs part-of-speech tagging with a Hidden Markov model. Parameters for these processes are estimated from a manually annotated corpus of currently about 1,500k words.[43] It is a freeware software available under a permissive license and standalone application (Hellwig, 2009).

[41] http://sanskrit.jnu.ac.in/post/post.jsp

[42] http://sanskrit.jnu.ac.in/post/post.jsp

[43] http://www.indsenz.com/int/index.php?content=sanskrit_tagger

- **Hindi POS-Taggers:**

In the year 2006, three different POS tagger systems were proposed for Hindi based on Morphology driven, ME, and CRF++ approach respectively. There have been already two attempts for parts of speech tagger developments in 2008 based on HMM approaches proposed by Shrivastava and Bhattacharyya. A Part of Speech Tagging for Hindi Corpus have been proposed by Nidhi and Amit Mishra in 2011. A POS tagger algorithm for Hindi was proposed by Pradipta Ranjan Ray, Sudeshna Sarkar, Harish V., and Anupam Basu.

- **POS-Taggers for Bengali:**

In the year 2007, two stochastic based taggers were proposed by Sandipan Dandapat, Sudeshna Sarkar and Anupam Basu using HMM and Maximum Entropy (ME) approaches. Further, Ekbal Asif developed a POS tagger for Bengali language using Conditional Random Fields (CRF++). In 2008, Ekbal Asif and Bandyopadhyay developed another machine learning based POS tagger using SVM algorithm. An Unsupervised Parts-of-Speech Tagger for the Bengali language was proposed by Hammad Ali in 2010. A Layered Parts of Speech Tagging for Bengali in 2011 was proposed by Chakrabarti from CDAC, Pune.

- **Tamil POS Taggers:**

Vasu Ranganathan proposed a Tamil POS tagger based on Lexical phonological approach. Another POS tagger was prepared by Ganesan based on CIIL Corpus and tagset. An improvement over a rule-based morphological analysis and POS Tagging in Tamil were developed by M. Selvam and A.M. Natarajan in 2009. Dhanalakshmi V., Anand Kumar, Shivapratap G., Soman K.P., and Rajendran S. of Amrita University, Coimbatore have prepared two parts of taggers for Tamil using their own developed tagset in 2009.

- **POS Taggers for Punjabi Language:**

Using the rule-based approach, a Panjabi POS tagger was developed by Mandeep Singh, Gurpreet Lehal, and Shiv Sharma in 2008. The fine–grained tagset contains around 630 tags consisting of all the tags for various classes of words, tags specific to some words, and tags related to punctuations. Only handwritten linguistic rules are used to disambiguate the POS information for a given word, based on the context information. Using the rule-based disambiguation approach, a database was designed to store the rules. Also, a separate database was maintained for marking verbal operator. The system reports an accuracy rate of around 80.29% including unknown words and 88.86% excluding unknown words.

- **POS Taggers for Telugu Language:**

NLP in the Telugu language is better off when compared with other South Dravidian and many other Indian languages. There are three noticeable POS taggers developments in Telugu, based on Rule-based, transformation-based learning, and Maximum Entropy-based approaches. An annotated corpus of 12000 words was constructed to train the transformation-based learning and Maximum Entropy-based POS tagger models. The existing Telugu POS tagger accuracy was also improved by a voting algorithm by Rama Sree, R.J. and Kusuma Kumari P in 2007.

- **POS Taggers for Malayalam:**

In 2009, Manju K., Soumya S., and Sumam Mary Idicula proposed a stochastic Hidden Markov Model (HMM) based part of speech tagger. A tagged corpus of around 1,400 tokens were generated using a morphological analyzer and trained using the HMM algorithm. The performance of the developed POS Tagger is about 90% and almost 80% of the sequences generated automatically for the test case was found correct. The second POS tagger is based on machine learning approach in which training, testing, and evaluation are performed with Support Vector Machine (SVM) algorithms developed by P.J. Antony, Santhanu P. Mohan and Dr. K.P. Soman of Amrita University, Coimbatore in 2010. They have proposed a new AMRITA POS tagset and based on the prepared tagset, a corpus size of approximately 180,000 annotated words were used for training the system. The performance of the SVM-based tagger achieves 94% accuracy and showed an improved result than HMM-based tagger.

- **POS Taggers for Kannada Language:**

P. J. Antony and K.P. Soman of Amrita University, Coimbatore proposed a statistical approach to building a POS tagger for Kannada language using SVM. They have proposed a tagset consisting of 30 tags. The architecture of the proposed POS tagger in the Kannada language is corpus-based and supervised machine learning approach. The POS tagger for the Kannada language was modeled using SVM kernel.

A corpus size of fifty-four thousand words was used for training and testing the accuracy of the tagger generators.

1.3.Why a Statistical POS Tagger for Odia?

This section contains various types of POS annotation, aims and scope of the research, hypothesis, rationale behind the selection of such a topic, and the uniqueness of the current study.

1.3.1. Parts of Speech Tagging

In corpus linguistics, POS-tagging is the "process of assigning a part-of-speech or other lexical markers to each word in a corpus" (Nainwani et al., 2004) or in other words, "the process of assigning to each word in a running text a label which indicates the status of that word within some system of categorizing the words of that language according to their morphological and/or syntactic properties" (Hardie, 2003).

"A Part-Of-Speech Tagger (POS Tagger) is a piece of software that reads text and assigns parts of speech to each word (and other tokens), such as noun, verb, adjective, etc." (Toutanova et al. 2003). The tagger assigns a (unique or ambiguous) POS tag to each token in the input and passes its output to the next processing level. Taggers can be categorized as rule-based, stochastic and hybrid.

Schachter in (1985) as cited in (Mitkov, 2003) has stated that parts of speech occur in all the natural languages and most valid criterion for deciding the categories of words is grammatical rather than semantic. The grammatical criteria to be considered are as follows:

- Syntactic distribution
- Syntactic function
- The morphological and syntactic classes that POS labels can be assigned to.

1.3.1.1.Rule-based POS Tagging

The rule-based tagger applies manually written rules to resolve issues while the stochastic tagger utilizes probability occurrences of words for a certain tag. E.Brill's tagger,[44] one of the first and most widely used English POS taggers which applies rule-based algorithms.

As opined by Brill (1992), the rule-based approach to machine learning and especially, parts of speech tagging is considered to overcome the limitations that are commonly attributed to the rule-based approaches to language processing. Those limitations are with in terms of robustness, and automatically acquisition of the human-encoded rules. In addition, he states that the rule-based tagger has many advantages over statistical taggers. They need for large-scale decrease in the information storage, the requirement of 'a small set of meaningful rules' as opposed to cumbersome statistics badly required for statistical parts of speech taggers, ease of 'improvements to the tagger', and 'better portability from one tagset or corpus genre to another' (Brill, 1992). But the rule-based taggers are not free from loopholes. They are 'non-automatic', quite expensive and 'time-consuming' (Megyesi, 1998).

1.3.1.2.Statistical POS Tagging

After getting the corpora labeled manually, automated POS annotation is conducted by machines applying different probability-based and context-based algorithms in computational linguistics. The statistical taggers have acquired an overwhelming accuracy without performing so much of any 'syntactic analysis on the input' (Brill, 1992). Brill (1992) has discussed that thousand lines of statistical information is required to encapsulate the contextual information of the tokens at hand. This information is usually encoded to the machine by way of tables of trigram statistics, "indicating for all tags tag^a, tag^b, and ranges the probability that tag^c follows tag^a and tag^b." The contextual information is addressed by fewer rules in the rule-based tagger. This feature of the tagger makes it more suitable "aiding in better understanding and simplifying further development of the tagger". Contextual information is provided by encoding in a precise and comprehensible manner. This precise and comprehensible 'representation of contextual information' is just as effective as 'the information hidden in the large tables of contextual probabilities' which can be observed from the comparison of error rates by the tagger.

[44] http://www.ling.gu.se/~lager/mogul/brill-tagger/

1.3.1.3.Hybrid POS Tagging

Finally, the hybrid tagger is an amalgamation of both the aforementioned approaches. There are limitations of statistical taggers; after a particular accuracy rate achievement, they need human encoded linguistic rules to perform much better.

1.3.1.4.Applications of the POS Taggers

Mitkov (2003) has discussed that a good parts of speech tagger can be applied as a pre-processor to several NLP applications. Large tagged corpora of higher linguistic reliability and quality are employed as data for research and development in the field of linguistics. Nouns and adjectives are used for indexing purposes in text indexing and for retrieval of relevant information in Information Retrieval (IR). Speech processing is also benefitted from POS tagging as the pronoun 'that' is pronounced differently from the conjunction 'that'.

1.3.2. Aims and Scope

For resource-rich languages like English, there are many electronically annotated available corpora of huge volume, for instance, WSJ[45] corpus, COCA,[46] British National Corpus,[47] The Time Magazine Corpus[48] and so on. Considering the NLP situation in ILs, they are poor so far as the availability of electronically annotated written corpora is concerned. One of the successful attempts has been to build a written corpus of around 100k sentences under the ILCI Project[49] in both the phases (see Fig. 1.), funded by the DIT, Govt. of India. Odia is a less resourced language as it is not empowered with sophisticated language technology and a considerable amount of corpus. Not much research work has been conducted in the field of developing language technologies. This envisaged proposed research work is the second of its kind in the field of statistical POS-tagging. The work aims at validating the ILCI Odia corpora and selecting an ideal statistical model for the said corpora out of the two models namely, CRF++ and SVM.

[45] https://catalog.ldc.upenn.edu/LDC2000T43
[46] http://corpus.byu.edu/coca/
[47] http://www.natcorp.ox.ac.uk/
[48] http://corpus.byu.edu/time/
[49] http://ildc.in/Oriya/Oindex.aspx

Objectives:

- The first and foremost objective is the automatic annotation of ILCI Odia POS corpora of approximately 300k tokens or around 50k sentences by one of the statistical parts of speech taggers after its accomplishment.
- The research envisaged here presents an experimental and comparative study between the CRF++ and SVM algorithms for the suitability of adaptation to Odia data. Furthermore, it discusses the issues and challenges based on the error types.
- The current study attempts to ensure the reliability and efficiency of the taggers by evaluating in a multi-modal format: qualitative and quantitative approaches. The models have been qualitatively evaluated by the Inter-Annotator Judgment and quantitatively or statistically evaluated by the evaluation tools of the models.
- The research aims at utilizing the statistical tagger for further research and development in the field of Odia computational linguistics such as in developing chunker, parser, machine translation and some other tools.
- The undertaken research aims at providing the NLP world a statistical parts of speech tagger to be available online publicly with an interface to cater to the demands in the community and serve as a beneficial stepping stone for further research and development.

1.3.3. Research Questions and Hypothesis

The current study hypothesizes that Support Vector Machines will perform better for Odia POS annotation corpora than the Conditional Random Fields because SVM has already been tested in English and some Indian languages and has been reported to have competitive accuracy.

The proposed study would attempt to address the following research questions:

- Which one (CRF++ or SVM) is the suitable and ideal model for Odia?
- How far does the tagger handle the issue of ambiguity and other subordinate linguistic issues?
- Which model, probabilistic or context-based model, functions well?
- How far does the tagger handle the issue of agglutination, prefixes, and suffixes?

1.3.4. Rationale for the Study

> "Odia is a morphologically rich language which possesses the salient features like PN and TAM being embedded in the verbs, serial verb constructions, ECV, causative constructions and conjunct verbs. Generally SOV word order is the most preferred one in sentence constructions while the possible word orders can be SVO and OVS constructions" (Jha, et al., 2014).

In other words, the language allows the scrambling process to undergo. So far as the agreement is concerned, verbs agree with their subjects on person and number. In Odia though the lexical gender is present, there is no grammatical gender; which has disappeared altogether in Bengali, Assamese, and Odia (Masica, 1991, pp.221). The said language is a nominative-accusative language with the possibility of non-nominative case construction (Patnaik, 2001).

Syntax-rigid languages like English do not allow the process of scrambling to take place, but the Indian languages like Hindi-Urdu (Kidwai, 2000) and Odia (Sahoo, 2010) and several other IA languages (Abbi, 2001, pp.28) undergo such processes of free-word-order and scrambling. The ILCI POS-annotated data has also embodied this feature of scrambling to render ambiguous POS constituents. When a human annotator tags the data, one is quite context-sensitive and takes into account all the factors like syntax, semantics and pragmatics for a particular sentence which is not true in the case of an automatic tagger.

Odia is a less computerized and less-resourced language although many tools have been developed. There are no huge annotated corpora completed by any group. Many of the tools developed till date are either not available online or not usable under a free license. The undertaken research aims at providing the NLP world a parts of speech tagger to be available online publicly with an interface to cater to the demands in the community and serve as a beneficial stepping stone.

The rationale for considering SVM and CRF++ is that both of them have already been applied to most of the Indian languages. Besides, SVM and CRF++ have already been applied to Hindi under ILCI Project which provides around 93 percent accuracy. The SVM model has been based on simple features like the medium verbose level, LRL mode and the rest of the features has been set to the default mode. On the other hand, unigram feature templates have been applied for the CRF++ model. The SVM is a binary classifier while the CRF++ is based on the probability. Therefore, one of the goals of the study is to seek for which of the models with the simple feature selection performs best for ILCI Odia corpora.

1.3.4.1.How is the Present Research Different from the Existing Ones?

In the ILCI project, POS tagging is being conducted for 17 ILs including Odia. The annotation work is conducted by a semi-automated online tool called ILCIANN (ILCI Annotation). In this work, ILCI-tagged Odia data has been verified and validated and two statistical models: (CRF++ -probability model) and (SVM- a binary classifier) have been trained. Thereafter, the test data set has to be run so as to measure the performance level of the two models. The ideal one has been selected for the rest of the data to be annotated in an automated manner. Then, the data set from the new domain has been provided and the output has been measured so as to ensure that the tagger learns and improves in the course of the training phase. The errors and mistakes have been addressed by way of precision inject of customized data to the tagger for its accuracy improvement. Finally, the errors from both the models evaluated in both the seen and unseen data have been discussed. Some of the ambiguous word forms and tags have also been discussed. Further, to check the reliability and the performance of both the models, they have been evaluated qualitatively by IA Agreement. The approach applied in the current study is based on the Data Approach. The amount of data utilized in the research figures approximately 236k, the amount of which has not been attempted so far in Odia and in most of the presently available taggers in ILs. To estimate the accuracy of the taggers, data from both the seen and unseen set has been provided. Generally, accuracy is measured on the basis of the output data provided to the taggers from the same training set of data.

The recently undertaken work is unique with respect to the volume of data used for developing the tools: 236k for the training phase and its half for the testing. Further, SVM tagger provides 96.85% and 93.59% respectively in the seen and unseen data whereas CRF++ tagger provides 94.39% and 88.87 respectively in both the said sets. This accuracy rate is competitively better than the existing Odia Neural Network POS tagger which provides 81% accuracy and the Odia SVM tagger which has 82%

accuracy. The present research is different from the existing taggers developed in ILs with regard to the availability of the huge volume supervised corpus from ILCI. Thus, the present study is limited to the application of the supervised corpus; although further experimental research can be conducted in the domain of unsupervised corpus.

1.4.Why a Computational Framework for Odia POS Tagger?

The POS-annotated corpus of a language has a number of significant applications as it can be employed in NLP applications like TTS (Text to Speech), information retrieval, word sense disambiguation, shallow parsing, information extraction, structural transfer, linguistic research for corpora and also as a foundation stone for advanced-level NLP tasks such as language parsing, phrase chunker, semantics, machine translation, speech recognition, online dictionaries and so on. Although two statistical taggers are existing, they provide a lower accuracy rate and are not available online. Therefore, the need of the hour is to develop a statistical tagger with a publicly accessible online user interface providing reliable and accurate output linguistically.

1.4.1. Understanding the Task

Before delving deep into any undertaken task, one needs to have an overall understanding of it. In this concerned research, the task encapsulates the preparation of statistical models; one of them to be best suitable for the language. For arriving at a particular decision and judgment, one needs to have an in-depth linguistic knowledge of the nuances of the given language and some computational understanding as to how the computer functions. Odia has some of the quite unique features; agglutination being one of them. This is one of the salient linguistic features of the Dravidian languages[50] also; although some of the Indo-Aryan languages such as Bengali and Marathi exhibit. Odia having agglutination as one of the salient features can be ascribed to the geographical location of the state; as it is situated in a belt where the Indo-Aryan languages converge. Besides, one also needs to have the computational aspect of the problem they are handling. The computer understands only the logic encoded through the binary code and not any human logicality to whichever extent the logic may be reasonable. In the present research, two computational models have been experimented and developed viz. the SVM and the CRF++. The former one is a classifier which

[50] Among the Dravidian languages Tamil is the most agglutinative language in comparison to other sister languages.

classifies the data and decodes the information whereas the latter is a ‘probability-based model’ which makes the probability of the input and provides the output based on its highest frequent probable tag for the given token. Finally, the best model functioning well for the language has been selected. This process has to undergo through several stages: training, testing and evaluation of the models. Therefore, it is indispensable that one needs to have both the knowledge before proceeding towards tackling the issues pertaining to the task one is assigned to.

1.4.2. A Brief Description of the Nature of the Problem and Solution

After understanding the task, one will proceed further to deal with the problem and propose a solution. This research deals with the problem of correct assignment of parts of speech labels to the corresponding words by the machine. The language-specific linguistic features create problems for NLP. In case of Odia, the features such as agglutination, compounding, morphophonemics, and unconventional orthography create disambiguation issues for the machine. Since these features play a crucial role in the language, the accuracy rate of a computational application solely depends upon how best one handles these features.

1.4.3. A Precise Introduction to the Statistical Models

This sub-section presents a precise introduction to the statistical models used for modeling the Odia POS taggers.[51]

1.4.3.1.Support Vector Machines (SVM)

In machine learning, support vector machines (Vapnik 1995, 1998) are supervised learning models with associated learning algorithms that analyze data and identify patterns that are applied in ‘classification and regression analysis’.

As has been put forth by Sober and Benedito (2001), (Cortes & Vapnik 1995, Edgar et al. 1997) have defined SVM as follows.

> “SVMs and other linear classifiers are popular methods for building hyper-plane-based classifiers from data sets and have been reported to have excellent generalization performance in a variety of applications. SVM is totally based on the statistical theory of learning developed by Vapnik (1995) and his team at AT & T

[51] For a complete description, please refer to section 4.2.

Bell Labs, which is a new learning algorithm and can be seen as an alternative training technique “for Polynomial, Radial Basis Function and Multi-Layer Perception classifiers.”

If a set of training examples is given, with each of them marked as coming from one of the two categories, an SVM algorithm used for training the data prepares a model “that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier”. The model built with SVM is a representation of the examples as ‘points in space’, mapped so that the instances of the separate categories are divided by a clear gap, making a hyperplane that is as wide as possible. Then, new instances are mapped into that same space and predicted to belong to a category based on which side of the gap they fall on: negative class or positive class.

1.4.3.2.Conditional Random Fields (CRF++)

CRF++ is a class of statistical modeling methods which is applied for the recognition of pattern, regression analysis, structure prediction and so on. Lafferty et al., (2001) observes “Conditional Random Fields offer a unique combination of properties: discriminatively trained models for sequence segmentation and labeling; combination of arbitrary, overlapping and agglomerative observation features from both the past and future; efficient training and decoding based on dynamic programming; and parameter estimation guaranteed to find the global optimum”. They are considered to be of ‘discriminative probabilistic undirected graphical model’ and applied for natural language applications or biological sequences (Lafferty et al., 2001), computer vision (He et al., 2004), shallow parsing, (Sha and Pereira, 2003), named entity recognition (Settles, 2004) and so on. They have been designed as an alternative probabilistic model to the Hidden Markov Model (HMM).

Singh et al., (2008) have stated that CRFs++ are applied for “calculating the conditional probabilities of values on designated output nodes given values on other designated input nodes of undirected graphical models”. A CRF++ is probability-based model as it can take the probability of each occurrence of the linguistic token into account while parsing sentences or annotating parts of speech.

1.4.3.3.Comparison between CRF++ and SVM

- CRFs++ are a class of statistical modeling methods while SVMs are supervised learning models with associated learning algorithms.
- CRFs++ are applied for pattern recognition and structured prediction. SVMs analyze data and identify patterns, applied in the 'classification and regression analysis.'
- The CRF++ was first modeled by (Lafferty et al., 2003) while SVM is based on the statistical learning theory developed by Vapnik (1995) and his team at AT&T Bell Labs.
- The CRFs++ are a type of 'discriminative undirected probabilistic graphical model' used for encoding known relationships between observations and construct consistent interpretations. The model is often utilized for labeling or parsing of sequential data, such as natural language text or biological sequences. On the other hand, if given a set of training examples, each marked as belonging to one of the two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other, by making it a non-probabilistic linear and binary classifier.
- So far as the efficiency in terms of speed and accuracy is concerned, the CRF++ can take a long time in processing while SVM takes less time in comparison to the CRF++. Besides, if one takes the accuracy into consideration, the CRF++ performs better even with a small data set applied for training including better feature selection. On the contrary, the SVM can function better with even no feature selection. But, the only thing required for that matter is a huge amount of data.

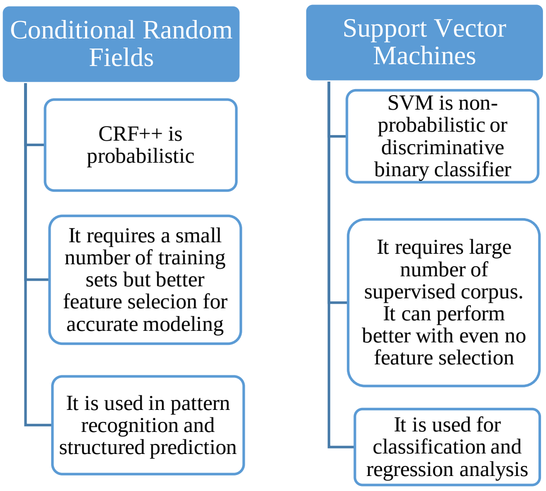

Conditional Random Fields
CRF++ is probabilistic
It requires a small number of training sets but better feature selecion for accurate modeling
It is used in pattern recognition and structured prediction
Support Vector Machines
SVM is non-probabilistic or discriminative binary classifier
It requires large number of supervised corpus. It can perform better with even no feature selection
It is used for classification and regression analysis

Fig. 2. Comparison of the CRF++ and SVM

1.5.Research Methodology

The methodology is important in any research as it lays the foundation for the succeeding stages. It encapsulates the methodology of data collection, interpretation and analysis. In other words, it contains the framework for the whole study.

1.5.1. Method of Data Collection

This sub-section contains the methodology adopted during for corpus collection and salient linguistic features of the data.

1.5.1.1.Corpus Collection

During the phase-I of the ILCI project (Banerjee et al., 2013 and Jha, 2011), 50k sentences corpora have been collected in Hindi and translated into 12 major Indian languages in the domains of health and tourism (Choudhary and Jha, 2011) including Odia. While collecting it has been ensured that the main domain involves almost every sub-domains. So under the domain like health, sub-domains such as, immune system, lungs and breathing, lifestyle, paediatrics, brain and nerves, reproductive systems, digestive system, endocrine system, blood, heart and circulation, eyes and vision and so on. Similarly, the sub-domains like pilgrimage, medical, adventure, space, war, culinary leisure, heritage ecotourism, mass travel and tours, disaster, dark, shopping, and nautical tourisms (Choudhary and Jha, 2011) under the tourism domain. In phase-II, the other five scheduled languages have been incorporated and the domains also covered entertainment, agriculture, religion, literature and so on with another 50k sentences collected corpora. The project includes translation and POS Tagging in phase-I whereas phase-II involves POS annotation, validation and chunking.

1.5.1.1.1. Salient Linguistic Features of the Data

There are many special features of the ILCI corpora: orthographic convention, incorporation of many words into the lexicon of Odia from Hindi and some linguistic structures from Hindi are few among them. Some of these sorts of features are probably due to the parallel first phase data which has been translated into the other ILs in the project from Hindi as the source language (SL).

- **Orthographic Convention:**

There are some parts of speech of which the orthographic convention varies under some circumstances. The postpositions are the most frequently and variedly used categories that directly or indirectly affect the POS judgment of the categories they are attached to. Similarly, many of the negative bound morphemes which are in fact attached to the free morpheme ‘be-verb auxiliaries’, are used separately from each other in orthography. As a result, it creates problems for the generally decided tags for the negative bound morphemes. In the example below, when the convention is followed the POS category of the word is a finite verb when it is not the category becomes a main verb with the addition of negative particle.

For instance

Conventional orthography: /heʊnɑhĩ/ ‘not getting done’ (V_VM_VF)

Violated orthography: /heʊ/ /nɑhĩ/ ‘not getting done’ (V_VM) (RP_NEG)

- **Loan Morphology from the SL:**

There are a large number of Hindi and Hindi-like morphological patterns loaned from the SL, i.e. Hindi. To instantiate a few examples, the Odia counterparts of Hindi words /sunəhəri/ ‘golden’, /kə̃ʈɪle/ ‘thorny’, /ʃɔkIn/ ‘fond of or interested’, /regisʈ̪ɑn/ ‘desert’ and so on are /sʊneli/, /kɔ̃nʈɑɟukʈ̪ɔ/, /sɔʊkɪɑ/ etc.

- **Acquisition of SL Linguistic Structures:**

There are a several examples from the corpora where especially the morphological and syntactic features have been directly incorporated into the target language (TL).

Examples like /məhət̪wəpurɳ bhʊmɪkɑ nɪbhɑɪ pɑrɪbɔ/ ‘you will play an important role’ instead of /mɔhɔt̪wɔpurɳɔ bhʊmɪkɑ ɡrɔhɔɳɔ kɔrɪbɔ/ ‘you will play an important role’, /lɔbreʐ khet̪ɔ/ instead of /purɳɔ khet̪ɔ/ etc. In both the examples although the English translations are the same, in their representation in Odia from the Hindi-like structure is quite different. In the former example, which is Hindi-like, the first two elements come under an NP while the last two elements come under a VP. Contrarily, the latter has an NP as the first two elements while the third one is another NP and the fourth one is the VP as the last two items will form a conjunct verb.

1.5.2. Method of Data Annotation

This sub-section contains the methodology adopted for the annotation of the Odia POS corpora collected earlier. The annotation work has been accomplished by using the ILCIANN App and following the BIS guideline.

1.5.2.1.BIS Tagset for Annotation under ILCI

The below demonstrated table figures the BIS superset tagset (see table below) for Odia. The BIS tagset is a hierarchical set designed by the POS Standardization Committee appointed by the Department of Electronics and Information Technology, Government of India. It contains the 11 top-level categories, their subtypes, 39 labels, annotation convention and examples in IPA. The top level category is the main grammatical category. The subtype contains all the types and subtypes of the main POS categories. The label 1 is the nomenclature used for the annotation convention while the labels of annotation convention are used for the annotation process.

	ILCI POS tagset for Odia under BIS				
Sl. No.	**Category**		**label**	**Annotation Convention**	**Examples of Odia in IPA**
	Top level	Subtype (lev-1)			
1	noun		N	N	
1.1		common	NN	N_NN	pɔt̪^{h}ɔrɔ, cɪnt̪ɑ, d̪ɔɭe etc.
1.2		proper	NNP	N_NNP	rɑmɔ, hɪmɑɭɔjɔ, etc.
1.3		verbal	NNV	N_NNV	k^{h}ɪɑ, pɔhɔ̃rɑ, nɔcɑ, k^{h}ɑɪbɑʈɑ

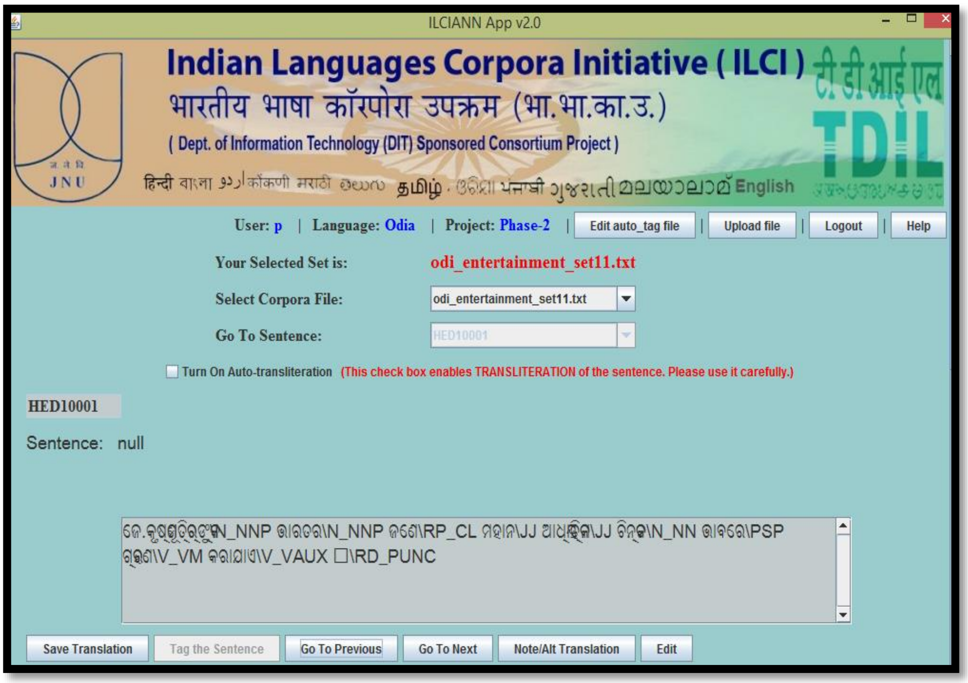
1.4		Nloc	NST	N_NST	ɑgɔku, pɔc^{h}ɔku, pɔre, pʊrbɔ, etc.
2	pronoun		PR	PR	
2.1		personal	PRP	PR_PRP	mũ, t̪ʊme, ɑpɔɳɔ etc.
2.2		reflexive	PRF	PR_PRF	nɪɟe, sʊɔjɔ̃ etc.
2.3		relative	PRI	PR_PRI	ɟɑharɔ, ɟɑhɑ̃kɔrɔ, ɟeũmɑnɔ̃kɔ
2.4		reciprocal	PRC	PR_PRC	pɔrɔspɔrɔ, ʊb^{h}ɔjɔ etc.
2.5		wh-word	PRQ	PR_PRQ	kɪe, kɑhɑrɔ, kɑhɑpɑĩ etc.
2.6		indefinite	PRI	PR_PRI	ɔnjɔrɔ, kɔuɳɔsɪ, kehɪ etc.
3	demonstrative		DM	DM	
3.1		deictic	DMD	DM_DMD	ehɪ, seɪ, ehɑ, t̪ɑhɑ, egʊɖɪkɔ, s egʊɖɪkɔ etc.
3.2		relative	DMR	DM_DMR	ɟeʊ̃gʊɖɪkɔ, ɟɑhɑrɔ
3.3		wh-word	DMQ	DM_DMQ	kɑhɑrɔ, kout̪hɪrɔ, keʊ̃gʊɖɪkɔrɔ etc.
3.4		indefinite	DMI	DM_DMI	ɔnjɔ, kɔuɳɔsɪ etc.
4	Verb		V	V	
		main	VM	V_VM	k^{h}ɑ, nɑc, d̪ek^{h} etc.
4.1.1			VF	V_VM	k^{h}ɑe, k^{h}ɑʊc^{h}ɪ, k^{h}ɑɪlɑ, k^{h}ɑut̪^{h}ɪlɑ, k^{h}ɑɪbɔ, k^{h}ɑut̪hɪbɔ etc.
4.1.2			VNF	V_VNF	k^{h}ɑɪ kɪ, nɑcɪ, etc.
4.1.3			VINF	V_VINF	k^{h}ɑɪbɑkʊ, k^{h}ɑɪbɑ pɑĩ, nɑcɪbɑ etc.
4.1.5			VNG	V_VNG	k^{h}ɑɪt̪^{h}ɪbɑ, k^{h}ɑʊt̪^{h}ɪbɑ etc.
4.2.1		auxiliary	VAUX	V_VAUX	ʊcɪt̪, d̪ɔrɔkɑrɔ, kɔrɪ, t̪^{h}ɪbɑ etc.
5	adjective		JJ	JJ	b^{h}ɔlɔ, ut̪t̪ɔmɔ, sʊ̃d̪ɔrɔ etc.

6	adverb		RB	RB	
7	Postposition		PSP	PSP	sɔ̃ge, paĩ, lagɪ etc.
8	conjunction		CC	CC	
8.1		Co-ordinator	CCD	CC_CCD	kahĩkɪ na, karɔɳɔ, ɟehet̪ʊ etc.
8.2		Subordinator	CCS	CC-CCS	ɟɔd̪ɪ t̪ebe, ɟet̪ebeɭe set̪ebeɭe, ɟe etc.
9	particles		RP	RP	
9.1		Default	RPD	RP_RPD	mɔd̪^{h}jɔ, hĩ, t̪ɔ etc.
9.2		Classifier	CL	RP_CL	goʈe, goʈa, k^{h}ɔnɖɔ etc.
9.3		Interjection	INJ	RP_INJ	bah, hɔ̃, hĩ etc.
9.4		Intensifier	INTF	RP_INTF	ɔt̪jɔnt̪ɔ, k^{h}ub, ɔt̪ɪ etc.
9.5		Negation	NEG	RP_NEG	na, nahĩ, nʊhɛ̃ etc.
10	quantifiers		QT	QT	
10.1		General	QTF	QT_QTF	ɔɭpɔ, besi, kɔm, ʈɪke etc.
10.2		Cardinal	QTC	QT_QTC	ekɔ, d̪uɪ, t̪ɪnɪ etc.
10.3		Ordinal	QTO	QT_QTO	prɔt̪^{h}ɔmɔ, d̪wɪt̪ijɔ, t̪rʊt̪ijɔ
11	residuals		RD	RD	
11.1		Foreign words	RDF	RD_RDF	languages of the other scripts
11.2		Symbol	SYM	RD_SYM	Mathematical and other symbols (%, $, <, >, (,), *)
11.3		Punctuation	PUNC	RD_PUNC	(, ; : ‘’ “” :- etc.)
11.4		Unknown	UNK	RD_UNK	Tags that are left undecided
11.5		Echo word	ECH	RD_ECH	bag^{h}ɔ-p^{h}agɔ, kɔʈa-c^{h}ɔʈa etc.

Table. 1. The Odia ILCI Parts of Speech Tagset under BIS

1.5.2.2.Online and Semi-automated Tool (ILCIANN APP v2.0)

Under the ILCI Project, 50k corpus from the phase-I has been annotated on the web ILCI[52] platform (see Fig. 1) manually. Some of the data have further been annotated by a semi-automated tool named ILCIANN App (Kumar et al., 2012) (see Fig. 3) manually. The tool has a special feature of ‘auto-edit tag list’ which automatically tags those tokens identical to the assigned token in the list (Nainwani et al., 2012). To increase the efficiency of the tool, one needs to tackle the auto-edit feature which may ably handle the semi-automated annotation. For tackling the tool ably, one needs to prepare a list of ‘lexical database’ of the close class categories such as nouns, adjectives and adverbs (Nainwani et al. 2012) which are language specific in nature.


ILCIANN App v2.0
Indian Languages Corpora Initiative (ILCI)
भारतीय भाषा कॉरपोरा उपक्रम (भा.भा.का.उ.)
(Dept. of Information Technology (DIT) Sponsored Consortium Project)
JNU
हिन्दी বাংলা कोंकणी मराठी తెలుగు தமிழ் ଓଡ଼ିଆ ਪੰਜਾਬੀ ગુજરાતી മലയാളം English
टी डी आई एल
TDIL
Edit auto_tag file
Upload file
Logout
Help
Tag the Sentence
Go To Previous
Go To Next
Note/Alt Translation
Edit
Save Translation

User: p | Language: Odia | Project: Phase-2

Your Selected Set is: odi_entertainment_set11.txt

Select Corpora File: odi_entertainment_set11.txt

Go To Sentence: HED10001

□ Turn On Auto-transliteration (This check box enables TRANSLITERATION of the sentence. Please use it carefully.)

HED10001

Sentence: null

ଜେ.କୃଷ୍ଣମୂର୍ତ୍ତି\N_NNP ଭାରତର\N_NNP ଜଣେ\RP_CL ମହାନ\JJ ଆଧ୍ୟାତ୍ମିକ\JJ ଚିନ୍ତକ\N_NN ଭାବରେ\PSP ଗ୍ରହଣ\V_VM କରାଯାଏ\V_VAUX □\RD_PUNC

Fig. 3. ILCIANN App v2.0

1.5.3. Method of Data Analysis

The analysis comprises of two major components: the evaluation of the tagsets for ILs, and the taggers evaluation. Under the tagsets evaluation, four prominent tagsets for ILs have been compared. Under the taggers evaluation, both mechanical and human evaluation have been conducted. The former has been conducted automatically whereas the latter has been done with the analysis based on an Inter Annotator Agreement of

[52] http://ildc.in/Oriya/Oindex.aspx

SVM and CRF++ outputs. Further, under mechanical evaluation, the accuracy per level of ambiguity, and parts of speech, ambiguity sets, and the unknown vs known, and the ambiguous vs unambiguous have been dealt with vividly along with the training, testing, and evaluation procedures for both the models. In addition, a detailed analysis of the errors has been made including discussions of the errors. Furthermore, linguistic disambiguation has been proposed to be employed for the lexical ambiguity sets along with two other approaches namely the data approach and the linguistic formulation of rules considering the salient linguistic features of the language.

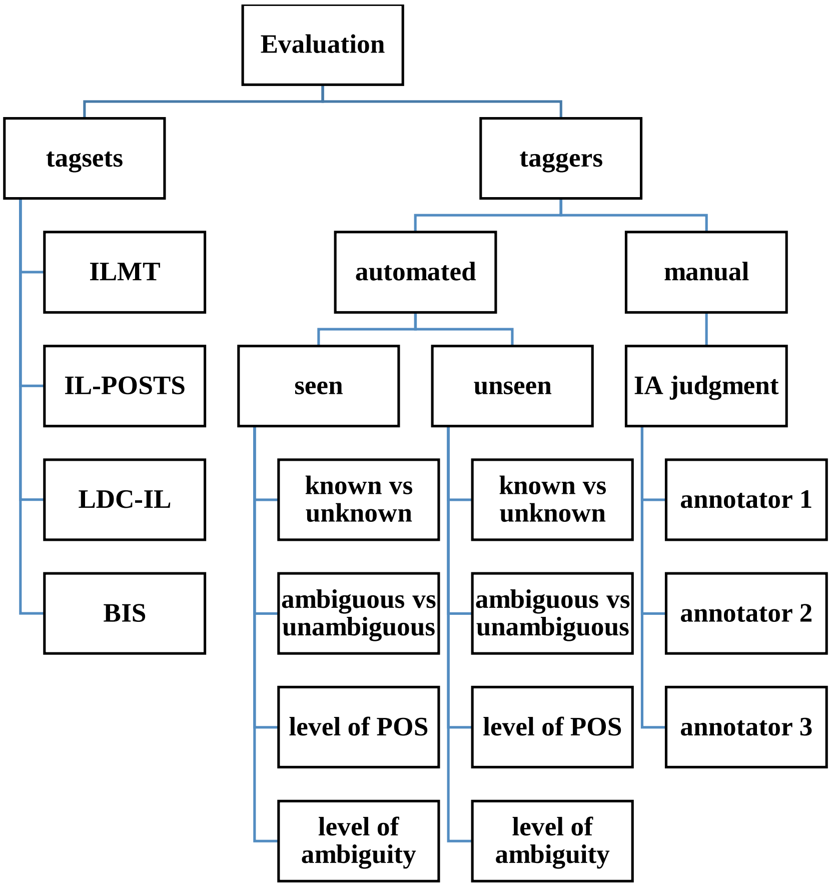


Fig. 4. Evaluation Diagram

CHAPTER 2

2. A DESCRIPTIVE SKETCH OF ODIA PARTS OF SPEECH CATEGORIES

This chapter encapsulates two sections: a precise introduction to Odia morphology with respect to the grammatical categories and a brief sketch of the Odia syntax in terms of phrases. The first section deals with the description of the eleven parts of speech whereas the second section provides an introductory background to different types of Odia phrases: noun, verb, adjective, and adverb.

2.1.A Brief Introduction to Odia Morphology

Indian languages have always been challenging for linguistics due to their diversity and multiplicity. India is the homeland for five language families: the Indo-Aryan, Dravidian, Austro-Asiatic, Tibeto-Burman, and the Andamanese (Abbi, 2001, pp. 24). Odia is a language, perhaps the only language, which contains both the IA and the Dravidian features. This reason could be ascribed to its geographical location as it is located in a belt where both the IA and Dravidian families converge. Besides, Boulton (2003) has stated that Odisha had been the homeland for the aboriginal and Dravidian tribes. There could be the possibility that Odia may have loaned some of the linguistic features from the Dravidian languages at that juncture period. Therefore, it can, however, be averred that Odia is a 'typologically-syntactically disturbed' IA language having both the IA and Dravidian features induced within itself (Patnaik, 2004). The Dravidian features that are observed in Odia are the occurrence of complementizer /bɒlɪ/ post-verbally, agglutination, 'not allowing participial agreement' and the curved shape of the alphabets etc.[53] The IA features to be considered are the relative-correlative construction, inflectional features, and many others.

The unmarked word order in Odia is also in line with the NIA word order, i.e. Subject-Object-Verb and free-floating constructions (Sahoo, 2010). Because of the reason of the fact that verbs agreeing with their subjects in person and numbers,[54] sentences can appear without the subject (Masica, 1991). This word order is usually

[53] Ibid.

[54] Note: there is no grammatical gender in Odia, but lexical gender is present.

adhered, but SVO and VSO constructions are also acceptable (Masica, 1991 and Abbi, 2001). Though Odia has 'relatively free word-order'; all the word orders are not correct and appropriate considering the notions of 'grammaticality, acceptability and correctness'.

In some eastern NIA languages, like Oriya, Sambalpuri, Bengali and Assamese, verb-less constructions (Abbi, 2001, pp. 28) are also possible without a verb and are grammatical, whereas Hindi-Urdu does not exhibit such features.

For example pɪlɑʈɪ bʰɒkɪlɑ

the boy hungry

'The boy is hungry'.

So there is a clear distinction contrastingly marked between the copular verbs and the existential verbs. Karpushkin in (1964, pp.91), as cited in Masica (1991), has stated that in Bengali and Odia the existential verbs sometimes have much to do with the 'durative aspect' of the sentence. /bʰɒkɪlɑ/ here is an adjective qualifying the noun and is existential in nature as it refers to the state of happiness of the topic in question. Odia verbs are inflected for tense and agree with their subjects in person and number, but not gender; as exemplified in the declension of the of the verb /jɑ/ 'go' in the present continuous tense, formed by combining the participle with inflected forms of the auxiliary /ʊɔcʰɪ/. Verbs agree with their subjects on persons and numbers. So, sentences can appear without subjects. Although Odia has the lexical gender, there is no grammatical gender.

case	Marker	meaning
nominative	Ø	null
Accusative	kʊ	to
Dative	kʊ	to
Instrumental	re/d̪ʋɑrɑ	with/by
Locative	re/ʈʰɑre	in/on/at
ablative	rʊ/tʰɑrʊ	from
genitive	rɔ	of/'s

Table. 2. Types of Case in Odia

As discussed by Debojit (2012) that Oriya has a rich case system; marking nominals for accusative/dative (-kʊ), instrumental (-re), ablative (-rʊ), genitive (-rɔ), and locative (-re/-rɪ) cases. Nouns in the nominative are not marked. Case markers may be preceded by plural markers /gɔcʰɔ-gʊɖɪkɔ-rɔ/, or by the definite marker. The accusative case is used only when the direct object is specific. Odia nouns can have both the forms, singular or plural: pilɑ ‘child’ pilɑ-mɑne ‘children’. In Odia, the case markers for dative, genitive, locative and ablative are /kʊ/, /rɔ/, /re or ʈʰɑre/ and /ʈʰɑrʊ/ respectively.

As discussed by Neukom (2003), Odia has a four-fold aspectual distinction- simple, imperfective, perfective and ingressive- which is seen in all five tenses and moods. There are two explicit aspect markers: The suffix /-ʊ/ marks imperfective aspect and the suffix /-ɪ/ marks perfective aspect. The third form, called simple, is formally unmarked for aspect. There could be another aspect added to the list, i.e. ingressive, which are particularly verbs that denote a process leading up to a state, whereas the state itself must be expressed by a verb form with its perfective aspect.

- **Odia Parts of Speech Categories:**

In both the oriental and occidental linguistic traditions, parts of speech have been recognized for a long time by scholars like Panini, Thrax, Priscian, and Varro. Dionysius Thrax has categorized eight classes of the word: noun, verb, particle, article, pronoun, preposition, adverb, and conjunction based on the formal criterion (Mitkov, 2003). According to Priscian, there are eight grammatical categories: noun, verb, participles, pronoun, preposition, adverb, interjection and conjunction.

In Odia, there are 11 broadly defined upper-level grammatical categories in the BIS scheme: noun, verb, pronoun, demonstrative, adverb, adjective, postposition, conjunction, particles, quantifiers and residuals.

2.1.1. Noun

Nouns are marked for the categories of number, gender, and case as suffixes (Neukom, 2003). Nouns are the categories that agree with the verbs by conjugation; case endings and classifiers agglutinate with them and with some other categories in Odia[55].

[55] Please refer to the section on agglutination (3.3.3.) for more information

During the annotation process, different types of nouns have been annotated as common nouns like, common, abstract, material, and collective for common noun (N_NN), while the proper names (no generic proper names; but only names referring to proper entities) have been tagged as proper (N_NNP). Verbal nouns have some of the features of the noun like definiteness, noun phrase acting like the agent and genitive case where the verbal noun is part of the possessive elements etc. (N_NNV). All the temporal and locational adverbs are annotated as locative nouns (be it the case that they appear as a single element or part of a complex postposition) (N_NST). There could be a huge number of words coming from other languages getting transliterated in the collected corpora.

Nouns having case markers attached to them:

Examples bɔɟɑrɔ-rʊ\N_NN ‘from the market’

Nouns having plural suffixes attached to them:

Examples brʊkʰjɔ-gʊɖɪkɔ\N_NN ‘trees’

Collective nouns:

Example pʰʊlɔ\N_NN penɖɑe\N_NN ‘a bouquet of flowers’

Transliterated words from other languages:

Examples ʈʊɪn\JJ sejɔrɪŋ\N_NN ɑd̪ʰɑrɔre\N_NN ‘on twin-sharing basis’ (the modifiers of the head noun are in English)

2.1.2. Pronouns

As has rightly been observed by Neukom (2003), personal pronouns refer to human beings. There are three persons in Odia; first, second and third. The second and the third persons show forms which contrast in honorificity. The second person distinguishes three degrees: familiar, polite and honorific, whereas the third person has only two forms: non-honorific and honorific. The pronouns can be changed into plural by the addition of a suffix /–mɑne/ for humans, or more rarely /sɔbʊ/ ‘all’ for the non-humans like the nouns in Odia. There is only pronoun for the first person plural, /ɑme/ ‘we’ (very rare /ɑme-mɑne/ and /ɑmbʰe/). There is a clear distinction between the inclusive and the exclusive verbs and with the agreeing pronouns. The case system of the pronouns is distinguished from the nouns in terms of the oblique case and nominative case. The pronouns have been categorized into six categories: personal, relative, reciprocal, reflexive, indefinite, and interrogative.

Pronouns	Person	Number	Genitive Case	Ablative Case	Dative Case
mʊ̃ ‘I’	first	singular	mɒ-rɔ ‘my’	mɒ-ʈʰɑrʊ ‘from me’	mɒ-t̪e ‘me’
ɑme or ɑmbʰe ‘we’	second	plural	ɑmɔ-rɔ ‘our’	ɑmɔ-ʈʰɑrʊ ‘from us’	ɑmɔ-kʊ ‘us’
t̪ʊ (informal) ‘you’	second	singular	t̪ɒ-rɔ ‘his’	t̪ɒ-ʈʰɑrʊ ‘from you’	t̪ɔ-t̪e ‘you’
t̪ʊme (-honorific) ‘you’	second	singular	t̪ʊmɔ-rɔ ‘your’	t̪ʊmɔ-ʈʰɑrʊ ‘from you’	t̪ʊmɔ-kʊ ‘you’
ɑpɔɳɔ (+honorific) ‘you’	second	singular	ɑpɔɳɔŋkɔ-rɔ ‘your’	ɑpɔɳɔŋkɔ-ʈʰɑrʊ ‘from you’	ɑpɔɳɔ-ŋkʊ ‘you’
se ‘he or she’	third	singular	t̪ɑ-rɔ/t̪ɑŋkɔ-rɔ ‘his or her’	t̪ɑ-rɔ/t̪ɑŋkɔ-ʈʰɑrʊ ‘from him or her’	t̪ɑ-kʊ ‘him or her’
semɑne ‘they’	third	plural	semɑnɔŋkɔ-rɔ ‘their’	semɑnɔŋkɔ-ʈʰɑrʊ ‘from them’	semɑn-ŋkʊ ‘them’

Table. 3. Pronoun and Case Markers

To differentiate pronouns (PRP) from demonstratives, they have been tagged as pronouns when their referents exist beyond the same clause boundary. In other words, if the referent is in the embedded clause, then the pronoun can come in the matrix clause. Otherwise, they are labeled as the demonstratives (DMD).

2.1.3. Demonstratives

There are two governing criteria that are followed for deciding the category of the demonstrtaives: the proximity and the distal. As put forth by Neukom (2003), the former are the proximal demonstrative /e/-series which refers to entities that are quite proximally located to the speaker, while the latter are the distal demonstratives /se/-

series that refer to the entities located quite far away. The phonetic elements /e/ and /se/ can be expanded by the suffix /-ɪ/, which forms /eɪ/ and /seɪ/ respectively. "The emphatic suffix /-ɪ/ can be inserted in many of the deictic forms of the /e/ and /se/-series, e.g. /seɪmɑne/ 'those', /eɪʈʰɑre/ 'here', /seɪʈʰɑre/ 'to that side', /seɪt̪ʰɪpɑĩ/ 'therefore', /semɪt̪ɪ/ 'like that' etc". The demonstratives have been categorized under four categories: the deictic, relative, indefinite, and interrogative.

Demonstratives	**distance**	**Number**	**Genitive case**	**Ablative case**	**Dative case**
ehɑ	proximal	singular	ehɑ-rɔ	ehɑ-ʈʰɑrʊ	ehɑ-kʊ
t̪ɑhɑ	distal	singular	t̪ɑhɑ-rɔ	t̪ɑhɑ-ʈʰɑrʊ	t̪ɑhɑ-kʊ
egʊɖɪkɔ	proximal	plural	egʊɖɪkɔ-rɔ	egʊɖɪkɔ-ʈʰɑrʊ	egʊɖɪ-kʊ
segʊɖɪkɔ	distal	plural	segʊɖɪkɔ-rɔ	segʊɖɪkɔ-ʈʰɑrʊ	segʊɖɪ-kʊ
eɪ	proximal	singular	eɪʈɑ-rɔ	eɪ-ʈʰɑrʊ	eɪʈɑ-kʊ
seɪ	distal	singular	seɪʈɑ-rɔ	seɪ-ʈʰɑrʊ	seɪʈɑ-kʊ

Table. 4. Demonstratives and Case Markers

2.1.4. Verbs

Verbs are inflected for person, number, tense and aspect whereas not for gender as there is no grammatical gender in Odia. Honorific plays a dominant role in making the verbs inflect. Therefore, verbs are also inflected for honorificity. Thus, they are marked for honorific with the persons: second (singular and plural) and third (singular and plural). There is no any honorific attached to both the singular and plural forms of the first person, and the second person plural is already an honorific pronoun. Both the singular and plural forms of the second person honorific mark the verb with the same marker. Third person singular /se/ 'he' denotes to both the feminine and masculine genders and also can be used both as honorific and non-honorific. In the third person plural /se-mɑne/ 'they', both of the honorific and non-honorific are ingrained in the same pronoun and verbs are marked for honorificity.

Syntactically, verbs determine the number and functions of noun phrase arguments in a sentence. Semantically, they express states, processes, and actions. The basic verbs, as well as causatives behave identically with respect to the aspect, mood, tense, and agreement features.

In Odia, the conjunctive participle marker is /kɪ/, but there is a possibility of the deletion of these markers. The nonfinite forms of the verbs behave like the conjunctive verbs. Instead, the perfectives, which are also finite forms of the verbs, function as the conjunctive participles of the verbs. Besides, the finite verb of the sentence is marked for PN and TAM features. In Hindi-Urdu, the case of conjunctive participle is slightly different from that of Odia with respect to the presence or absence of the participle markers. Here in the Hindi language, for a verb to be a conjunctive participle candidate, it has to have the markers like /kər/ and /ke/ obligatorily. The finite verb seems to bear the inflections for all the TAM features and PN markers.

In Odia, generally there are verb groups quite complicated to reach an agreed opinion as to decide what their labels are when there is the standard convention adhered. To decide the category of the verbs occurring as part of a verbal group is a daunting task as most of the linguistic information like TAM and PN features are ingrained in them. But when the orthographic convention is different, it is quite easy to decide the category of at least the finite and non-finite verbs. All the infinitival constructions have been annotated as infinitives (V_VM_VINF) having the infinitive markers like /bɑ/ and /-bɑ-kʊ/. The category of finite verbs (V_VM_VF) has been decided on the basis of TAM features. For instance, any of the verbs inflecting for either of the tense, aspect and mood are to be annotated as finite and those which are not to be tagged as per their category based on the context they are in. Those verbs that are marked as non-finite (V_VM_VNF) have the inflectional endings like /ɪ/ and /kɔrɪ/ after the stem. Verbs which have the progressive and perfective participle markers and appear before nouns qualifying them are tagged as a gerund (V_VM_VNG). Main verbs (V_VM) are the verbs having the root or stem as part of their verb group whereas the auxiliaries (V_VAUX) are the modals, be verbs, some of the vectors, and conjunctive participle markers occurring as single entities.

2.1.5. Adjectives

“Adjective is a member of the word class whose main function is to specify an attribute, characteristic, etc. of a noun phrase” (Brown, 2006).

Adjectives (JJ) are the words that qualify or modify nouns: be it in the attributive or in the predicative position or occurring as part of the conjunct verb. Adjectives can also function as the subject of the sentence where there is no overt noun as the subject (Sahoo, 1996).

For example:

- As the subject of the sentence:

  lɑlʈɪ mɒt̪e d̪ɔrɔkɑrɔ

  red-one me need

  "I need the red one".

- Attributive Position:

  mɪʈʰɑ pʰɔɭɔ

  'sweet fruit'

- Predicative Position:

  pɪlɑʈɪ bʰɑrɪ sʊnd̪ɔrɔ ɔʈe

  boy the very beautiful is

  "The boy is quite handsome".

- As part of the conjunct verb:

  se sɒsɪlɑ ɔʈe

  he thirsty is

  "He is quite thirsty".

2.1.6. Adverb

Adverbs are those parts of speech that qualify or modify a verbal phrase and an adjectival phrase.

"Adverb is a member of the word class whose characteristic function is to specify the manner in which the action of a verb is performed" (Brown, 2006).

In the annotation process, the temporal and the locational adverbs have been labeled as the spatio-temporals (N_NST) and have included the manner adverbs only under the category of adverbs (RB). In addition, as per our guidelines the temporal and the frequentative and resultative adverbs like, /sɔrʊɔd̪ɑ/, /et̪^{h}ɪpɑĩ/, /t̪eɳukɔrɪ/ etc. have been tagged as adverbs even though they are not manner adverbs. But, quite interestingly, in most of the ILs, the manner adverbs at the morphological level seem to be different from what it appears to be at the phrase level. At these instances, the level of the tag has been decided on the basis of the Phrase Structure rules and they have been annotated as adverbs; so as in other cases.

2.1.7. Postpositions

Postpositions are the POS that follow a noun phrase either as case markers or as postpositions, or as part of a complex postposition. One has to identify the distinctions between case and case markers. Because, in some languages, they could be confusing; as alternately being used. The former is a semantic relation while the latter is a demonstration of that very relation in terms of the phonological word (Abbi, 2001, pp. 127). One has to further take into account that the postpositions and case markers are often interchangeably used. Broadly speaking, postpositions can be divided into two categories: the simple and the complex. The former consists of only one word while the latter comprises more than one.

Neukom (2003) has classified the Odia postpositions into four groups: ‘true postpositions, lexicalised verbal forms, relational nouns, and bare nouns or adjectives’.

2.1.8. Conjunctions

Haspelmath (2000) has defined coordinations, as cited in Abbi (2001), as

> “Syntactic constructions in which two or more units of the same type are combined into a larger unit and still have the same semantic relations with other surrounding elements”.

There are three types of coordination in most of the IA languages: co-ordinating, sub-ordinating and relative-correlative constructions.[56]

[56] Ibid

> “Subordination is a type of syntactic linking in which one linguistic unit is dependent on another” (Brown, 2006).

While, on the other hand, the subordinating conjunctions (CC_CCS) are those parts that subordinate two linguistic elements. While annotating, one can obviously face issues like how to label the correlative clause-like structures of demonstratives and pronouns where the personal or deictic occur in either of the clause and their relative counterparts occur in either of the other clauses.

2.1.9. Particles

So far as particles are concerned, most of them function like adverbs, but they can not occur independently as they do not have the status of independent words. Thus, they require a host to appear in sentences. There is clearly marked distinction between the focus and modal particles. The modal particles refer to the expression of the speakers’ perspective and quite rare in the standard usage of the language. On the other hand, they can be frequently used informally by the speakers in the spoken discourses (Neukom 2003).

As averred by Neukom (2003), there are three important particles that are frequently used in the informal communication: /lɒ/, /re/ and /be/. They are used “to express the relationship of familiarity between the speaker and hearer and are restricted to utterances where the addressee is referred to by the 2nd person singular familiar /t̪ʊ/. Their position is sentence final; so that their scope is the whole sentence, or they are attached to a proper noun or to a title” (Neukom, 2003).

So far as the particles are concerned, they do not have a particular meaning and function as discourse markers, emphasis, negation, and exclamation. Nouns having the markers like /ʈe/, /ʈɪ/, /ʈɑ/, /ɟɔɳɔ/, and /k^{h}ɔ̃ɖɔ/ have been tagged as the classifiers (RP_CL). They occur as parts of the noun phrase making the head noun a host. At often times, the infinitive verbs also have the inflections like /ʈɑ/ after the infinitive constructions which is a general feature for a classifier-marked noun. For example, the verb /k^{h}ɑɪbɑ-ʈɑ/. In these cases, the superseding category of the word has been given priority i.e. infinitive verb. Interjections (RP_INJ) such as /bɑh/, /hɔ̃/, /hĩ/, /sɑbɑs/ are used to refer to the exclamatory nature of the sentence. The POS categories that intensify the manner of the work being conducted and the quality of the nouns are known as intensifiers (RP_INTF). In other words, the categories that intensify the adjectives and adverbs are called as intensifiers. In Odia, the categories like the quantifiers, when preceded by the adjectives and adverbs, are intensifiers as for example, /bɔhut̪/ in the occurrence /bɔhut̪ sund̪ɔrɔ/, /k^{h}ub/ in /k^{h}ub begɔre/ etc. Even, the words like quantifiers when preceded by other quantifiers are tagged as intensifiers; for instance, /bɔhut̪ kɔm/.

2.1.10. Quantifiers

As defined by (Neukom, 2003),

> "Quantifiers determine the quantity of their (following) head when used in attributive position. They may determine the quantity of the set they refer to when used as nominal. Most of the quantifiers occur as intensifying particles as well".

Words like /ɔɭpɔ/, /besɪ/, /kɔm/, /ʈke/ etc. are labeled as the general quantifiers (QT_QTF), cardinals are numerals both written in number or words. Ordinals are the words referring to a particular place of order in a sequence.

2.1.11. Residuals

Residuals have been divided into five sub-categories: foreign words, symbols, punctuations, unknown words, and echo words.

Foreign words (RD_RDF) are the words written in other scripts. For example, the words from English are not foreign unless written in Roman script in the Odia data. Thus, if they are written in Odia, they are tagged as according to the tags that they should have; based on their categories within a given context. All the mathematical and scientific symbols like the brackets, ampersand, currency symbols, addition, subtraction etc. are labeled as symbols (RD_SYM). The tokens like the comma, colon, semicolon, quotation marks etc. are tagged as the punctuations (RD_PUNC). The categories that are left undecided, be it a word from the language or from a foreign language, are tagged as unknown words (RD_UNK). The categories such as /bɑg^{h}ɔ-p^{h}ɑgɔ/, /kɔʈɑ-c^{h}ɔʈɑ/ are annotated as the Echo words (RD_ECH). The second part of reduplicated expressions is also to be marked as echo words while the first part of the phrase is labeled based on its category.

2.2. A Brief Introduction to the Odia Syntax

The current sub-section contains an introduction to Odia syntax in nutshell at the level of phrase using phrase level constituents.

2.2.1. Noun Phrase

As discussed by Kalyan Malini Sahoo (1996) in her thesis submitted to the CIEFL, numerals, classifiers, quantifiers, articles can occur both pre and post-nominally within an NP/DP without the meaning being affected.

- cɑrɪ-ʈɑ p^{h}ɔɭɔ

  four-CL fruits

  ‘four fruits’

- p^{h}ɔɭɔ cɑrɪ-ʈɑ

  fruits four-CL

  ‘four fruits’

- cɑʊɭɔ bɔst̪ɑe

  rice sack-CL

  ‘a sack of rice’

- bɔst̪ɑe cɑʊɭɔ

  sack-CL rice

  ‘a sack of rice’

DPs are further grammatical and meaningful without having an overt noun within them.

For Example nɑlɪɑ-ʈɑ mɒt̪e d̪ɔrɔkɑrɔ

red-CL I need

‘I need the red one’.

As stated by Sahoo (1996), one of the characteristics of Odia DP/NP is that the number “is realized either as nominal or as nominal inflection but not as both”.

- d̪ɔs-ʈɪ gɔcʰɔ

  ten-CL trees

  ‘ten trees’

- gɔcʰɔ-gʊɖɪkɔ

  ‘tree-pl’

  ‘trees’

- *d̪ɔs-ʈɪ gɔcʰɔ-gʊɖɪkɔ

  ten-CL tree-pl

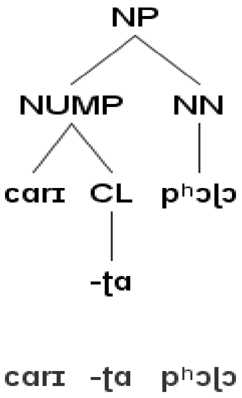
  ‘ten trees’

- *gɔcʰɔ d̪ɔs-ʈɪ gʊɖɪkɔ

  tree ten-CL pl

  ‘ten trees’

Odia /ekɔ/ has been derived from the Sanskrit numeral one and it is too formal. Therefore, it is not used in the speech. Instead, the words like /e-/, /ɡoʈɑe/, /ɡɒʈɪe/, /ɡoʈe/, and /-kɔ/. Thus, the order of sequence of the elements within the DP/NP is numeral+classifier+noun.

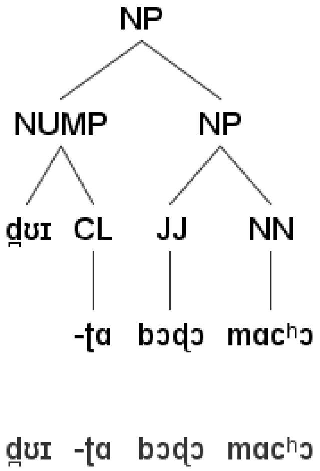


As exemplified by Sahoo (1996) and (Bharti et al., 2006), If an adjective qualifies a noun within the DP/NP, it occupies the premodifier place. Sahoo has also stated that the specifiers’ place can be filled by the numerals or numerals+classifiers occurrence.

For example d̪ʊɪʈɑ bɔɖɔ mɑcʰɔ

Two-CL big fish

‘Two big fishes’.

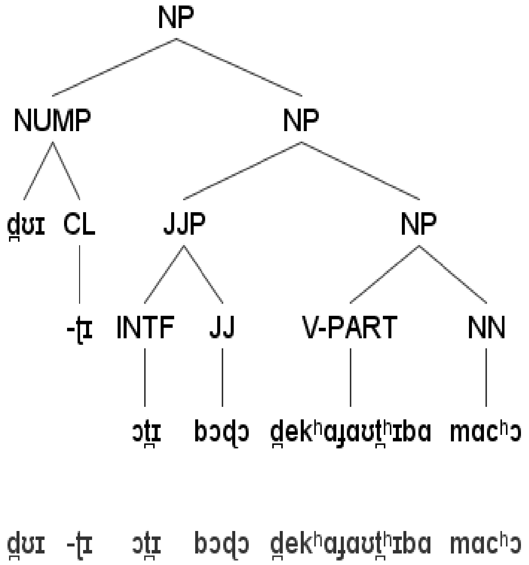


The imperfective participial verb preceded by an adjectival phrase can also occur under the DP/NP.

For example d̪ʊɪ-ʈɪ ɔt̪ɪ bɔɖɔ d̪ekʰɑɟɑʊt̪ʰɪbɑ mɑcʰɔ

two-CL very good looking fishes

‘two very good looking fishes’.



Similarly, an adverbial phrase can precede a participial (imperfective and perfective) verb and both modify the following noun as premodifiers. In Odia a noun phrase can have these elements as mentioned in the following rules with these combinations of grammatical categories.

NP = QT_QTC + RP_CL + RP_INTF + JJ + V_VM_VNG + N_NN

NP = QT_QTC + N_NN

NP = QT_QTC + RP_CL + N_NN

NP = JJ + N_NN

NP = RP_INTF + JJ + NN

NP = V_VM_VNG + N_NN

For example ɔt̪jɔnt̪ɔ bʰɔlɔ bʰɑbɔre kɔrɑɟɑɪt̪ʰɪbɑ kɑmɔ

very well-done work

‘really well-done work’

2.2.2. Verb Phrase

In English, the verbal phrase can contain all the different forms of verbs like the present/past simple, present/past participle, auxiliaries, and modals etc.

VP = VB/VBD/VBN/VBG/VBZ/VBP

Similarly, the different forms of verbs: the main, auxiliary, infinitive, finite, non-finite, gerundive etc. are under the verbal phrase in Odia. The following combinatory classes are possible for verbal phrase.

VP = V_VM + V_VAUX

VP = V_VM_VINF + V_VM_VF

VP = V_VM_NF + V_VM_VF

VP = V_VM + V_VM_NF + V_VM_VF

VP = JJ + V_VM_VF

VP = N_NN + V_VM_VF

The adverbs can also function as the premodifier to the verbs. Kachru (2006) has divided the verb phrases into two types in Hindi: the simple and the complex.

The simple verb phrase contains a finite verb with the inflections like TAM (tense-aspect-mood) and PN (person-number) etc. on the contrary, the complex verbal phrase consists of the complements, adverbial and objects. Similar constructions are also feasible in Odia with each of the respective categories.

2.2.3. Adjectival Phrase

Koul (2008) has categorized the adjectival phrases of Hindi into two broad categories: simple and complex. Further, he has sub-categorized the simple adjectives into derived and non-derived. Non-derived adjectives are the basic types of adjectives like /bʰɔlɔ/, /sʊnd̪ɔrɔ/, /cʰɒʈɔ/ etc. The derived adjectives are the types that are derived from some of the other parts of speech categories: say, adverbs. One of the instances of the derived adjective is that which is derived commonly from the adverbs /pɑkʰɔ/ /d̪urɔ/ etc.

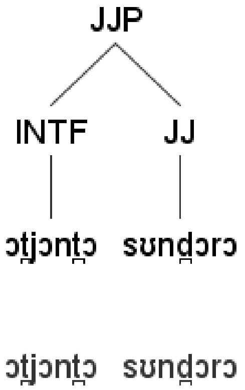
For example JJP = RP_INTF + JJ

JJP = RB + JJ

JJP = JJ

For example ɔt̪jɔnt̪ɔ sʊnd̪ɔrɔ

'quite handsome'



The complex adjectival phrases are either finite or non-finite. The finite are full relative clauses while the non-finite are participial forms of verbs used as adjectives.

2.2.4. Adverbial Phrases

An adverbial phrase can consist of a preceding intensifier and a following postposition.

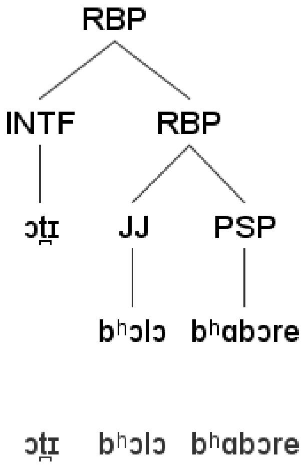
For instance RBP = RP_INTF + RB/JJ + PSP

RBP = RB + PSP

RBP = RB

For example ɔt̪ɪ bʰɔlɔ bʰɑbɔre

‘quite properly’



Koul (2008) & Abbi (1980) have stated that verbs can be reduplicated in Hindi to make the adverbs. Koul states that adverbs are reduplicated to intensify and stress the situation. However, they are not mandatory in any construction.

For example kʰɑɪ kʰɑɪ se ɑsɪlɑ

eating eating he came

“He came while eating”.

Manner adverbs can also be preceded by the intensifiers. The temporal and the spatial nouns are also adverbs, but they have been included under the broader category of nouns. Neukom (2003) has specified that the adverbial phrase in odia can have the occurrence of adverbs followed by post-positions.

For instance se ʈʰɪk bʰɑbɔre kɑmɔ kɔlɑ

he proper way work did

“He did the work properly”.

CHAPTER 3

3. PARTS OF SPEECH GUIDELINES FOR ANNOTATING ODIA CORPORA UNDER THE BIS SCHEME

This chapter presents four prominent sections. The first one lays emphasis on providing a list of annotation labels used for the annotation of the corpora along with their corresponding abbreviated tags while the second one deals with the parts of speech description along with their corresponding labels. The following section contains the issues and challenges while annotating parts of speech using BIS tagset under the ILCI. The final section proposes solutions for annotation scheme.

The whole ILCI (Indian Languages Corpora Initiative) corpus has been annotated based on the Bureau of Indian Standard (BIS) annotation scheme developed by the Department of Electronics and Information Technology (DeitY), Govt. of India with some modifications under the ILCI Project. The scheme is a hierarchical set with a total of 39 annotation labels common to all Indian languages under the ILCI.

3.1.List of Annotation Labels with Corresponding Parts of Speech

The present section contains the annotation labels and their corresponding parts of speech alphabetically. While annotating, the tag for quotative compound expressions (CC_CCS_UT) has been done away with.

	Annotation labels	parts of speech
1.	CC_CCD	coordinating conjunction
2.	CC_CCS	subordinating conjunction
3.	DM_DMD	deictic demonstrative
4.	DM_DMI	indefinite demonstrative
5.	DM_DMQ	interrogative/wh demonstrative
6.	DM_DMR	relative demonstrative
7.	JJ	adjective
8.	N_NN	common noun
9.	N_NNP	proper noun
10.	N_NST	spatio-temporal noun

11. N_NNV	verbal noun
12. PSP	postposition
13. PR_PRC	reciprocal pronoun
14. PR_PRF	reflexive pronoun
15. PR_PRI	indefinite pronoun
16. PR_PRL	relative pronoun
17. PR_PRP	personal pronoun
18. PR_PRQ	interrogative/wh pronoun
19. QT_QTC	cardinal quantifier
20. QT_QTF	general quantifier
21. QT_QTO	ordinal quantifier
22. RB	adverb
23. RD_ECH	reduplicative echo word
24. RD_PUNC	punctuation
25. RD_RDF	foreign word
26. RD_UNK	unknown word
27. RD_SYM	symbol/special character
28. RP_CL	classifier
29. RP_NEG	negation
30. RP_INJ	interjection
31. RP_INTF	intensifier
32. RP_RPD	default particle
33. V_VAUX	auxiliary verb
34. V_VM	main verb
35. V_VM_VF	finite verb
36. V_VM_VINF	infinitive verb
37. V_VM_VNF	non-finite verb
38. V_VM_VNG	gerundive verb

3.2.Parts of Speech Description with Their Corresponding Annotation Labels

This section contains the description of the parts of speech in a detailed manner along with examples from the data used in developing the Statistical Odia Parts of Speech Tagger. They are as follows alphabetically.

3.2.1. Adjective-JJ

Adjective has not been sub-divided into any further subcategories. There is one category for an adjective which is self-explanatory. The adjectives can be attributive, predicative, and gradable.

The compounds, that are hyphenated and function as the modifiers or qualifiers to a noun attributively, have been tagged as adjectives.

For example   prɔkrʊt̪ɪ-bɪkɑrɔ-sunjɔ\JJ sehɪ\DM_DMD ɑnɔnd̪ɔ\N_NN

The attributive and predicative adjectives in agglutinating forms have been annotated as adjectives.

For example   sund̪ɔri\JJ ɟʰɪɔ\N_NN ‘beautiful girl’

kɑbjɔ\N_NN sɔbdɑrt̪ʰɑɭɔnkɑrɑd̪ɪbɔhʊɭɔ\JJ (sɔbdɑrt̪ʰɔ+ɔɭɔnkɑrɔ+ɑd̪ɪ+bɔhʊɭɔ) ‘the epic is full of word-meanings’

The suffix endings like /-bʰukt̪ɔ/, /-hinɔ/, /-siɭɔ/, /-rɔhɪt̪ɔ/, /-ɟɔnɪt̪ɔ/, /-rɒd̪ʰɔkɔ/, /-d̪ɑjɔkɔ/, /-mɔjɔ/, /-bɔhʊɭɔ/ etc. are generally added to adjectives as suffixes and thus tagged as adjectives wherever they have been found.

The comparative and superlative endings of the adjectives are also tagged as adjectives.

For example   prɪjɔ-t̪ɔrɔ-t̪ɔmɔ\JJ ‘dear-er-est’

brʊhɔt̪-t̪ɔrɔ-t̪ɔmɔ\JJ ‘big-er-est’

ʊccɔ-t̪ɔrɔ-t̪ɔmɔ\JJ ‘high-er-est’

If they are parts of the conjunct verbs:

For example   bɔncɪt̪ɔ\JJ hɒɪcʰɪ\V_VM_VF ‘deprived of’

kʰʊsɪ\JJ helɑ\V_VM_VF ‘got happy’

Transliterated words from other languages:

For example   sɪŋl\JJ sɔplɪmenʈ\N_NN (English) ‘single supplement’

kɔnʈɪle\JJ bʊd̪ɑ\N_NN (Hindi) ‘thorny bush’

Feminine morphological adjectives:

For example sɔkt̪ɪsɑɭɪni\JJ d̪ebi\N_NN ‘powerful deity’

bʰɔd̪rɑ\JJ mɔhɪɭɑ\N_NN ‘polite lady’

3.2.2. Adverb-RB

Adverbs are those parts of speech that usually qualify or modify a verbal phrase and adjectives. In the annotation process, the temporal and the locational adverbs have been labeled as the (N_NST) and have been included the manner adverbs only under the category of adverbs (RB). In addition, as per our guidelines, the temporal and the frequentative and resultative adverbs like, /sɔrʊɔd̪̃ɑ/, /et̪ʰɪpɑĩ/, /t̪eɳukɔrɪ/ etc. have been tagged as adverbs even though they are not manner adverbs. But quite interestingly, in most of the Indian languages the manner adverbs at the morphological level seem to be different from what it appears to be at the phrase level. At these instances, the decision has been taken in favour of the Phrase Structure Rules and they have been tagged as adverbs.

The manner adverbs are under the category of adverbs.

For instance gɑɖɪ\N_NN ɑrɑmre\RB cɔɭɑɔ\V_VM_VF “Drive the vehicle easily”.

bʰɔlɔ\RB bʰɑbɔre\RB kɑmɔ\N_NN kɔrɔ\V_VM_VF “Do the work well”.

When they are used as followed by a cardinal.

For instance pɑkʰɑpɑkʰɪ\RB egɑrɔ\QT_QTC hɔɟɑrɔ\QT_QTC ʈɔnkɑ\N_NN ‘approximately 11,000 rupees’

Temporal:

For instance ɑɟɪ\RB ɑmɔkʊ\PR_PRP reɳʊkɑrʊ\N_NNP ɑgɔkʊ\N_NST ɟɪbɑrɔ\V_VM_VINF ɔcʰɪ\V_VM_VF

“Today, we have to go beyond Renuka”.

Frequentative:

For instance se\PR_PRP sɔrbɔd̪ɑ\RB kɑmɔ\N_NN kɔre\V_VM_VF

“He always does work”.

Resultative:

For instance et̪ʰɪpaĩ\RB t̪akʊ\PR_PRP kʰɔrapɔ\JJ lagʊcʰɪ\V_VM_VF

“Therefore, he is feeling bad”.

3.2.3. Auxiliary Verb-V_VAUX

The auxiliaries (V_VAUX) are the modals indicative of different moods (many of them have been tagged as finite verbs as they inflect for the tense and aspect).

For instance bɪsesɔ d̪ʰjanɔ d̪eba\V_VM_VINF d̪ɔrɔkarɔ\V_VAUX

‘Need to pay special attention’

nɔ\RP_NEG kɔrɪba\V_VM_VINF ʊcɪt̪\V_VAUX

‘Should not do’

Some of the ‘be’ verbs (many of them have been tagged as finite verbs as they inflect for the tense and aspect):

For instance t̪ɔrɔɭʊ\V_VM t̪ʰɪbarʊ\V_VAUX

‘Because of melting’

Some of the vector verbs in a compound verb construction:

For instance bʰɔrt̪ɪ\V_VM kɔrɪ\V_VAUX d̪eba\V_VM_VINF d̪ɔrɔkarɔ\V_VAUX

‘Should be admitted’

In an occurrence with a series of verbs:

For instance kɔrɪ\V_VM parɪ\V_VAUX nɔt̪ʰɪlɪ\V_VM_VF

‘I could not do’

3.2.4. Cardinal Number-QT_QTC

The cardinal quantifiers are absolute numbers, either in digits or in words such as ୧, ୨, ୩, ୪, ୫, ekɔ, d̪ʊɪ, t̪ɪnɪ, carɪ, pancɔ etc. In other words, all the cardinal numbers are tagged as cardinal quantifiers irrespective of their writing convention; whether in numeral or in the word. Cardinals with classifiers: goʈɪe, d̪ʊɪʈɪ, t̪ɪnɒʈɪ etc. have been tagged as classifiers.

Absolute numbers:

Example ekɔ\QT_QTC d̪ʊɪ\QT_QTC ‘one, two’

୧ ୨, ୩୪୫\QT_QTC ‘12, 345’

Absolute cardinals denoting uncertain numbers:

Examples lɔkʰje\QT_QTC ‘a lakh’

nɪjʊt̪e\QT_QTC ‘a million’

kɒʈɪe\QT_QTC ‘a crore’

3.2.5. Common Noun-N_NN

Different types of nouns have been annotated as common nouns such as common, abstract, material, and collective. For example /pɪlɑ/ ‘boy’, /pɔt̪ʰɔrɔ/ ‘rock’, /cɪnt̪ɑ/ ‘thought’, /d̪ɔɭe/ ‘group’ etc. Besides, all the words, capable of forming an argument of the verb, can possess the case markers, and are followed by the post-positions in the language, have been tagged as common nouns.

Nouns having case markers attached to them:

Examples bɔɟɑrɔ-rʊ\N_NN ‘from the market’

sɔrirɔ-re\N_NN ‘in the body’

brʊkʰjɔ-rɔ\N_NN ‘of the tree’

d̪ɪnɔ-kʊ\N_NN ‘to the day’

Nouns having plural suffixes attached to them:

Examples brʊkʰjɔ-gʊɖɪkɔ\N_NN ‘trees’

hɑt̪ɪ-mɑne\N_NN ‘elephants’

Collective nouns:

Example pʰʊlɔ\N_NN penɖɑe\N_NN ‘a bouquet of flowers’

Nouns with classifier markers:

Examples pɪlɑ-ʈɪ/ʈɪe/ʈɑrɔ/ʈɪrɔ\N_NN ‘the boy, of the boy, the boy’s’

Transliterated words from other languages:

Examples ʈʊɪn\JJ sejɔrɪŋ\N_NN ɑd̪ʰɑrɔre\N_NN ‘on twin-sharing basis’ (the modifiers of the head noun are in English)

regɪst̪ɑn\N_NN ɔncɔɭɔ\N_NN ‘desert area’ (the modifier of the head noun is in Hindi)

3.2.6. Coordinating Conjunction-CC_CCD

Coordinating conjunctions are those parts of speech that conjoin two linguistic elements of equal status; be it words, phrases, or clauses. Words like /kɪmbɑ/ ‘or’, /kɪnt̪u/ ‘but’, /ɒ/ ‘and’, /ɔrt̪ʰɑt̪/, /bɑ/ ‘or’, /t̪ɔt̪ʰɑ/ are typical coordinators.

For example /kɑhĩki nɑ/ ‘because of which’, /kɑrɔɳɔ/ ‘because’, /ɟehet̪u/ ‘because of or since’

Conjoining two words:

For example sɔpʰɑ\JJ ɒ\CC_CCD sʊɑst̪ʰjɔbɔrd̪ʰɔkɔ\JJ pɔbɔnɔ\N_NN ‘clear and healthy air’

Conjoining two phrases:

For example mɒrɔ\PR_PRP bʰɑɪrɔ\N_NN mɪkɪ\N_NNP mɑʊs\N_NN ɒ\CC_CCD t̪ɑrɔ\PR_PRP ɖɒremɔn\N_NNP

‘My brother’s Micky Mouse and his Doreman’

Conjoining two clauses:

For example t̪ʊri\N_NN bɑd̪ɔnɔ\N_NN sɑrɑ\DM_DMI d̪esɔre\N_NN gunɟɑjɔmɑnɔ\JJ hɒɪʊtʰɪt̪ʰɑe\V_VM_VF kɑhĩkɪnɑ\CC_CCD ehɑkʊ\DM_DMD set̪ebeɭe\RB pɒlɪs\JJ reɖɪɒre\N_NN prɔsɑrɪt̪ɔ\N_NN kɔrɑɟɑɪt̪ʰɑe\V_VM_VF |/RD_PUNC

“Trumpet playing starts echoing all across the country because it is broadcast in the Polish Radio then”.

Conjoining items in a list:

For example ehɪ\DM_DMD t̪ʰɔrɔ\N_NN ɑme\PR_PRP bɔd̪rɪnɑt̪ʰɔ\N_NNP ,\RD_PUNC ked̪ɑrɔnɑt̪ʰɔ\N_NNP ,\RD_PUNC ɟɔmʊnɒt̪ri\N_NNP ebɔŋ\CC_CCD ehɑrɔ\DM_DMD ɑgɔre\N_NST t̪ʰɪbɑ\V_VM_VNG kɪcʰɪ\QT_QTF st̪ʰɑnɔgʊɖɪkɔ\N_NN d̪ekʰɪbɑkʊ\V_VM_VINF ɟɑʊt̪ʰɪlʊ\V_VM_VF | RD_PUNC

“This time we had gone to see Badrinath, kedarnath, Jamuntri, and some places situated just beyond that”.

3.2.7. Classifier-RP_CL

“A classifier, sometimes called a measure word, is a word or morpheme used in some languages to classify the referent of a countable noun, according to its meaning. In languages that have classifiers, they are often used when the noun is being counted or specified (i.e., when it appears with a numeral or a demonstrative)” (ILCI, 2010). “The classifiers mainly occur either as proper classifiers, attached to numerals or to the quantity word kete ‘how many; some’, or as indefinite markers, in combination with the suffix /-e/” (Neukom, 2003). The classifier markers are /-ʈɑ/, /-ʈI/, /ɟɔɳɔ/, and /kʰɔɳɖɔ/. They can occur with all types of nouns, cardinals, ordinals, demonstratives, and verbal nouns.

Nouns:

lɒkɔʈɪ ‘the man’ pɪlɑʈɪ ‘the boy’

Cardinals:

Examples gɒʈɪe\RP_CL ʈɪm\N_NN ‘one team’

sɒhɔɭɔ\QT_QTC ɟɔɳɔ\RP_CL ‘six people’

cɑrɪʈɪ\RP_CL mjɑc\N_NN ‘four matches’

Ordinals:

Examples prɔt̪ʰɔmɔ\QT_QTO kʰɔnɖɔ\RP_CL ‘the first piece’

prɔt̪ʰɔmɔʈɑ\RP_CL ‘the first one’

Demonstratives:

eɪʈɪ\RP_CL ‘this one’ seɪʈɪ\RP_CL ‘that one’

Classifier reduplication:

Examples gɒʈɪ\RP_CL gɒʈɪ\RP_CL kɔrɪ\V_VM_VNF ‘making one by one’

kʰɔnɖɔ\RP_CL kʰɔnɖɔ\RP_CL kɔrɪ\V_VM_VNF ‘making piece by piece’

However, one must be careful as the classifiers get attached with the verbal nouns and general quantifiers that have not been marked as classifiers.

3.2.8. Default Particle-RP_RPD

Particles are those parts of speech that have no meaning in isolation and they compose meaning when attached with another part. The common default particles are /hĩ/, /bɪ/, /mɔd̪ʰjɔ/, /t̪ɔ/ etc. in Odia.

Examples kehɪ\PR_PRI t̪ɔ\RP_RPD nɔ\RP_NRG rɔkʰɪle\V_VM_VF ‘also nobody kept me’

cʰɑɟɑ\N_NN mɔd̪ʰjɔ\RP_RPD ɑnekɔ\QT_QTF ‘the shadow is also much’

ehɑ\DM_DMD d̪ʋɑrɑ\PSP hĩ\RP_RPD ‘by this only‘

3.2.9. Deictic Demonstrative-DM_DMD

In Hindi, “The deictic demonstratives are default demonstratives that demonstrate the noun it modifies. The deictic demonstratives in Hindi are typically /jəh/, /ʋəh/, /je/ and /ʋe/. These always occur before the noun they modify” (ILCI, 2010). To differentiate pronouns (PR) from demonstratives (DM), they have been tagged as pronouns when their referents exist beyond the same phrase boundary. In other words, if the referent is in the preceding phrase, then the pronoun can come in the other following phrase. Otherwise, they are labeled as the demonstratives.

Examples jəh\DM_DMD ʂəhər\N_NN bɔhət̪\RP_INTF prɑcin\JJ hɛ\V_VM “This city is quite old”.

ʊs\DM_DMD gʰər\N_NN ki\PSP cʰət̪\N_NN pəkki\JJ hɛ\V_VM “The roof of that house is cemented”. (ILCI, 2010)

The demonstratives like the following:

Examples ehɪ\DM_DMD st̪ʰɔɭɔ ‘this place’

sehɪ\DM_DMD prɔkɑrɔ ‘that way’

egʊɖɪkɔ\DM_DMD hɑt̪ɪmɑnɔnkɔrɔ sɔrirɔ “These are the elephants’ bodies.”

segʊɖɪkɔ\DM_DMD sɔhɪt̪ɔ ‘with those’

When the noun does not follow immediately but the noun is an inanimate being:

Example eɦɑ\DM_DMD ekɔ\QT_QTC cʰɒʈɔ\JJ pɑhɑɖɪɑ\JJ bɔst̪ɪ\N_NN “this is a small hilly hamlet”.

When the noun is not present within the same clause boundary, but the noun has to be compulsorily an inanimate being:

Example t̪ɑhɑ\DM_DMD pɔre\N_NST ‘after that’

3.2.10. Finite Verb-V_VM_VF

The category of finite verbs (V_VM_VF) has been decided on the basis of TAM features. For instance, verbs (irrespective of its canonical grammatical categories) inflecting for either of the tense, aspect, and mood have been annotated as finite and those which are not are tagged as per their category based on the context they are in.

When the writing convention of the compound verb is a single linguistic string and the copular verb is joined with the main verb.

Example ɑʊʈ\N_NN kɔrɪd̪eɪt̪ʰɪle\V_VM_VF ‘got someone out’

When the negative bound morpheme /nɔ/ is infixed with the verb.

Example kɔrɪbɑkʊ\V_VM_VINF bʰʊlɪnɔt̪ʰɪle\V_VM_VF “he did not forget to do.”

When the modal is inflecting for the person, number, tense, and mood features.

Example d̪ekʰɑd̪eɪ\V_VM pɑre\V_VM_VF

When the existential verbs play the role of the finiteness of the clause in the absence of any other verb.

Example st̪ʰɪt̪ɔ\JJ ɔcʰɪ\V_VM_VF ‘is situated or present’

mɑt̪ɑ\N_NN mɔnd̪ɪrɔ\N_NN ɔʈe\V_VM_VF “it is the temple of the goddess”.

3.2.11. Foreign Word-RD_RDF

“Only those words which are written in a foreign script should be marked as foreign word, even if the annotator understands the foreign script” (ILCI, 2010).

For instance,

The foreign words like ‘human’ written in Roman script or the word ‘पेड़’ in the Devanagari script, only when they are written in different scripts other than the Odia script, have been tagged as foreign words.

Examples Human\RD_RDF (English)

पेड़\RD_RDF (Hindi) ‘tree’

The words from other languages transliterated into Odia have been tagged according to their corresponding categories. For instance, the same words like ‘ହ୍ୟୁମାନ’ /hjʊmɑn/ and ‘ପେଢ଼’ /peɽ/ have been tagged as adjective or noun and noun respectively.

3.2.12. Gerundive Verb-V_VM_VNG

Verbs which have the progressive /ʊt̪ʰɪbɑ/ and perfective participle /ɪt̪ʰɪbɑ/ markers and appear before nouns qualifying them are tagged as a gerund.

Progressive or imperfective participle:

pɑɳɪre\N_NN rɔhʊt̪ʰɪbɑ\V_VM_VNG pɔsʊ\N_NN ‘water-dwelling animal’

When the main verb is separated from the following progressive participle:

lɔgɑʊt̪ʰɪbɑ\V_VM_VNG lɒkɔ\N_NN ‘fixing man’

Perfective participle:

ɑsɪt̪ʰɪbɑ\V_VM_VNG ʈrekɪŋ\N_NN d̪ɔɭɔgʊɖɪkɔ\N_NN ‘arrived trekking teams’

Existential:

gɔrbʰɔre\N_NN t̪ʰɪbɑ\V_VM_VNG sɪsʊ\N_NN ‘pride-bearing child’ or ‘the child who is in pride’

3.2.13. General Quantifier-QT_QTF

The general quantifiers do not indicate any precise quantity, e.g. /t̪ʰɒɖɑ/ ‘a little’, /bɔhət̪/ ‘a lot of’, /zɟɑd̪ɑ/ ‘much’, /kʊcʰ/ ‘some’, /kɔm/ ‘less’ etc.

However, one has to keep in mind that some of the general quantifiers are also used as intensifiers when they are followed by adjectives immediately. “Whenever quantifiers occur with nouns (either following or preceding), it could be general quantifiers” (Nainwani et al, 2012). Further, some can also be used as personal pronouns. Thus, one has to consider the contextual linguistic information while annotating. The most commonly used general quantifiers in Odia are /ɔnekɔ/ ‘many’, /ɔd̪ʰɪkɔ/ ‘much’, /ɔd̪ʰɪkɑ̃sɔ/ ‘most’, /kebɔɭɔ/ ‘only’, /ɪt̪jɑd̪ɪ/ ‘et cetera’, /ɔd̪ʰɪkɔt̪ɔrɔ/ ‘most’, /ɑhʊrɪ/ ‘more’, /kʰʊb/ ‘very’ etc.

Canonical general quantifiers used for quantifying the quantity of the nouns:

Examples ɔnekɔ\QT_QTF ɟɑgɑre\N_NN ‘in many places’

ɔd̪ʰɪkɔ\QT_QTF ɟɑgɑre\N_NN ‘in many places’

ɔd̪ʰɪkɑ̃sɔ\QT_QTF sɑt̪ʰɪ\N_NN ‘most of the mates’

kebɔɭɔ\QT_QTF bɔrsɑrʊ\N_NN ‘only from the rain’

ɪt̪jɑd̪ɪ\QT_QTF ‘et cetera’

ʈjɑksɪ\N_NN ɑd̪ɪre\QT_QTF ‘taxi etc.’

ɔd̪ʰɪkɔt̪ɔrɔ\QT_QTF hɒʈel\N_NN ‘most of the hotels’

When followed by cardinal quantifiers:

Examples mɑt̪rɔ\QT_QTF d̪ʊɪ\QT_QTC ɟɔɳɔ\RP_CL ‘only two people’

When followed by verb:

ɑhʊrɪ\QT_QTF bɔɖʰɪɟɑɪt̪ʰɑe\V_VM_VF ‘is increased much’

ʊbʰɔjɔ\QT_QTF rɑɟɔkijɔ\JJ reɭɔgɑɖɪgʊɖɪkɔ\N_NN ‘both the royal trains’

Cardinals when used as general:

For example sɔhɔ\QT_QTF sɔhɔ\QT_QTF hɑʈʊɑ\N_NN ‘hundreds of market venders’

Canonical intensifiers used as general quantifiers:

For example kʰʊb\QT_QTF pɔsɔnd̪ɔ\N_NN ‘much like’

3.2.14. Indefinite Demonstrative-DM_DMI

Like for indefinite pronouns, the indefinite demonstratives refer to unspecified objects, places or things. These words are /kɪsi/, /kɒɪ/, /kəhĩ/, /kəbʰi/ etc. in Hindi. Similarly, in Odia there are indefinite demonstratives like /kɔʊɳɔsɪ/ ‘any’, /ɔnjɔ/ ‘other’ etc. As for instances in Hindi,

kɪsi\DM_DMI d̪ɪn\N_NN ʋəh\PR_PRP ɑegɑ\V_VM “on any day he may come.”

kɒɪ\DM_DMI ləɖkɑ\N_NN ɑjɑ\V_VM “some boy came.”

For example in Odia,

kɔʊɳɔsɪ\DM_DMI sɔmɔsjɑ\N_NN ‘any problem or which problem’

ɔnjɔ\DM_DMI kɑrɟjɔ\N_NN ‘other work’

3.2.15. Indefinite Pronoun-PR_PRI

The indefinite pronouns refer to unspecified objects, places or things. These words are /kɪsi/, /kɒɪ/, /kəhĩ/, /kəbʰi/ etc. in Hindi.

Examples sɔmɔst̪e\PR_PRI bʰɔkt̪ɪ\N_NN ‘all persons do devotion’

kehɪ\PR_PRI ɟɔɳe\RP_CL ‘someone’ or ‘anyone’

prɔt̪jekɔ\PR_PRI lɒkɔ\N_NN ‘each person’

3.2.16. Infinitive Verb-V_VM_VINF

Infinitives are often preceded by /t̪ɒ/; but not necessarily. There are Indian languages in which the demarcation between the infinitival verb and gerundial verb is blurred in Hindi. The canonical forms of the infinitives in Odia has been taken into consideration while annotating the whole corpus of the Odia language.

Functioning as a gerundive construction:

bʊlɪbɑrɔ\V_VM_VINF mɔɟɑ\N_NN ‘pleasure to move’

Clear infinitive case:

pɔcɪbɑkʊ\V_VM_VINF d̪ebɑ\V_VM_VINF ʊcɪt̪\V_VAUX ‘should allow to be rotten’

Functioning as a conjunct verb:

sɔnd̪ʰɑnɔ\N_NN kɔrɪbɑ\V_VM_VINF ɟɑɳɪt̪ʰɪle\V_VM_VF ‘he knew how to search’

Functioning as an agglutinative construction:

sʊkʰɪ\V_VM ɟɪbɑrʊ\V_VM_VINF (ɟɪbɑ+kɑrɔɳɔrʊ ‘because of getting dry’)

Augmenting a new clause:

pɑɳɪ ɔnt̪ɪre cɑlɪɟɪbɑ\V_VM_VINF ɟɒgʊ̃\PSP ʈjʊb\N_NN bɑhɑrɔ\V_VM kɔrɑɟɑɪt̪ʰɑe\V_VM_VF “tube is taken out because of water getting into the stomach.”

3.2.17. Interjection-RP_INJ

Interjections are particles which denote exclamatory utterances. The common exclamatory marks in Hindi are/ ɑʔ/, /hɑj/, /ʊpʰ/ etc. in Hindi. The common exclamatory marks in Odia are /ɑhɑ/, /ɒhɒ/, /ɑre/, /ɑhe/, /hɑjɔ/, /bʰɒ/ etc.

3.2.18. Intensifier-RP_INTF

Intensifiers are words that intensify the adjectives or adverbs. The common intensifiers in Hindi are /behəd̪/, /ət̪jənt̪/, /bəhʊt̪/ etc. in Hindi. Similarly, in Odia there are some specific words that can be both used as the intensifier and general quantifier: /ɔt̪jɔnt̪ɔ/, /bɔhʊt̪/, /ɔt̪ɪ/, /kʰʊb/, /ɔtjɔd̪ʰɪkɔ/ etc.

Intensifying adjectives:

ɔt̪jɔnt̪ɔ\RP_INTF sʊnd̪ɔrɔ\JJ ‘most beautiful’

bɔhʊt̪\RP_INTF bʰɔlɔ\JJ ‘very good’

ɔtjɔd̪ʰɪkɔ\RP_INTF rɒmɑncɔkɑri\JJ ‘much adventurous’

Intensifying adverbs:

ɔt̪ɪ\RP_INTF ɑscɔrɟjɑnʊɪt̪ɔbʰɑbɔre\RB ‘very surprisingly’

kʰʊb\RP_INTF begɔre\RB ‘quite fast’

Intensifying the quantity of things:

bes\RP_INTF ɔd̪ʰɪkɔ\QT_QTF kʰɑd̪jɔ\N_NN ‘quite a lot food’

ɔt̪ɪ\RP_INTF besɪ\QT_QTF pɔsɔnd̪ɔ\N_NN ‘very much like’

3.2.19. Interrogative/wh Demonstrative-DM_DMQ

The wh-demonstratives are the same wh-words (or question words) which act as wh-pronouns. The difference is that in their demonstrative function, they do not ask a question, rather only demonstrate. The wh-word demonstratives are /kɒi/, /kɪsi/, /kɔn/ etc. in Hindi.

keʊ̃\DM_DMQ kɑmɔ\N_NN ?\RD_PUNC ‘which work?’

kɒʊgʊɖɪkɔ\DM_DMQ nebɔ\V_VM_VF ?\RD_PUNC “which things will you take?”

kɔɳɔ\DM_DMQ kɔrɪbɔ\V_VM_VF ?\RD_PUNC “what will you do?”

3.2.20. Interrogative/wh Pronoun-PR_PRQ

The interrogative pronouns are the pronouns that are used to ask questions.

kɪe\PR_PRQ kɔhɪbɔ ?\RD_PUNC “who will speak?”

ehɑ kɑhɑrɔ\PR_PRQ gʰɔrɔ ?\RD_PUNC “whose house is this?”

kɔɳɔ\PR_PRQ kʰɑɪbɔ ?\RD_PUNC “what will you eat?”

kebe\PR_PRQ ɟɪbɔ ?\RD_PUNC “when will you go?”

3.2.21. Spatial and Temporal Noun-N_NST

Spatio-temporal nouns can be used with reduplicative and agglutinative forms.

Commonly used:

eʈʰɪ\N_NST ‘here’

seʈʰɪ\N_NST ‘there’

pɔre\N_NST ‘after’

eɳe\N_NST ‘helter’

kʰjɔɳɪ\N_NST ‘as soon as’

With agglutination:

eʈʰɑre\N_NST ‘at here’

seʈʰɑre\N_NST ‘at there’

purbɔrʊ\N_NST ‘from before’

ɑgɔkʊ\N_NST ‘to the front’

Reduplicated:

eɳe-t̪eɳe\N_NST ‘helter-skelter’

pɔcʰe-pɔcʰe\N_NST ‘behind-behind’ (in the sense of following)

3.2.22. Main Verb-V_VM

Main verbs (V_VM) are the verbs having the root or stem as part of their verb group whereas the auxiliaries (V_VAUX) are the modals, be verbs, some of the vectors, and conjunctive participle markers occurring as single entities.

Adjective to verbal derivation:

sɔmbʰɔbɪ\V_VM pɑre\V_VM_VF ‘may happen’

bʰɔleɪ\V_VM heɪ\V_VM_VNF ‘showing good’

After the non-finite verb:

kɑnd̪ɪ\V_VM_VNF kɑnd̪ɪ\V_VM_VNF kʰɑʊ\V_VM t̪ʰɪlɑ\V_VM_VF “he was eating by crying.”

In a serial verb occurrence:

kɔrɪ\V_VM d̪eɪ\V_VAUX rɔkʰɑ\V_VM ɟɑɪcʰɪ\V_VM_VF “it has been kept after getting completed.”

3.2.23. Negation-RP_NEG

In Odia, negation is one of the most important parts of speech as it can occur independently as in place of the finite verb of the declarative sentence, as an infix morpheme in a verbal occurrence and occurs with the verb.

Negative infix morpheme as negative:

kɔrɑɟɑɪ-nɔ-t̪ʰɪlɑ\V_VM_VF ‘had not been done’

Occurs individually:

kʰɑɪ\V_VM nɑhĩ\RP_NEG ‘has not eaten’

If it occurs with the verb:

neɪnɑhĩ\V_VM_VF ‘has not taken’

3.2.24. Non-finite Verb-V_VM_VNF

Those verbs that are marked as non-finite (V_VM_VNF) have the inflectional endings like /i/ and /kɔrɪ/ after the stem.

Part of the compound verbs with both elements separated:

pɪɪ\V_VM kɔrɪ\V_VM_VNF ‘having drunk’

Part of the compound verbs with both elements joined:

mɪsɑɪkɔrɪ\V_VM_VNF ‘having mixed’

As a form of causation:

lɔgɑɪbɑrʊ\V_VM_VNF ‘because of adding’

As part of the conjunct verb expression:

sɑhɑsɔ\N_NN kɔrɪ\V_VM_VNF ‘having dared’

Scrambled order of verbal occurrence:

d̪inɔkrʊsnɔ\N_NNP bɒle\V_VM_VF ehɑ\DM_DMD bʰɑbɪ\V_VM_VNF “Dinakrushna said having thought of this”

Reduplicated expressions:

kʰɑɪkʰɑɪ\V_VM_VNF ‘by eating’

kʰɑʊkʰɑʊ\V_VM_VNF ‘eating’

gɔcʰɔ\N_NN ʊʈʰɪ\V_VM_VNF ʊʈʰɪkɑ\V_VM_VNF ‘the tree moving up and down’ pʰʊlɔ\N_NN pʰʊʈɪ\V_VM_VNF pʰʊʈɪkɑ\V_VM_VNF ɟɑʊt̪ʰɪbɔ\V_VM_VF ‘the flower blossoming’

3.2.25. Ordinal Quantifier-QT_QTO

The ordinals refer to the order part of the numeric digits such as /pəhəlɑ/, /d̪usrɑ/, /t̪isrɑ/ etc. in Hindi. Some of the ordinals in Odia also inflect for gender and also take classifiers along with them.

For instance prɔt̪ʰɔmɑ\QT-QTO pʊt̪ri\N_NN ‘first daughter’

sɒɖɔsi\QT-QTO ɟʰɪɔ\N_NN ‘sixteen-year girl’

3.2.26. Personal Pronoun-PR_PRP

In Hindi, personal pronouns cover all the pronouns that denote to person, place or thing. This includes all their cases as well: for example mɛ/, /həm/, /merɑ/, /həmɑrɑ/, /mʊɟʰe/, /həmẽ/, /mʊɟʰi/, /həmĩ/, /t̪ʊm/, /t̪ʊmhɑrɑ/, /t̪ʊmʰẽ/, /t̪ʊɟʰi/ etc.

3.2.27. Postposition-PSP

Postpositions are the parts of speech that follow a noun phrase. They may sometimes get attached to the nouns in the forms of case markers if they are simple and can occur separately if they are part or whole of the complex postpositions. However, one has to take into consideration the cases where the postpositions function like independent nouns and can become the arguments of the verbs. In addition, one has to be quite sure about the distinction between the sets of the locative and temporal nouns, and the postpositions.

Case markers as postpositions:

gɒlɑpɔ ʈʰɑrʊ\PSP t̪ɔɭe ‘below than rose’

Postpositions occurring independently:

mɒ\PR_PRP pɑĩ\PSP ‘to or towards me’

t̪ɑnkɔ\PR_PRP ɔnʊsɑre\PSP ‘according to or as per him’

ehɑkʊ\DM_DMD cʰɑɖɪ\PSP ‘except this’

ehɪ\DM_DMD bɪsɔjɔre\PSP ‘about this’

semɑnɔnkɔ\PR_PRP prɔt̪ɪ\PSP ‘to them’

When /bɑlɑ/ construction is attached with the noun, they are tagged as a noun. When it occurs independently, is tagged as PSP.

bɔhʊpɔt̪nɪ\N_NN bɑlɑ\PSP ‘a person with many wives’

/bɑlɑ/ occurring with the verbs has been tagged as per category of the type of verbs it is attached to.

d̪ʰʊmrɔpɑnɔ\N_NN kɔrɪbɑbɑlɑ\V_VM_VNG mɔhɪɭɑmɑnɔnkʊ\N_NN ‘to smoking women’

In the above example, the /bɑlɑ/ is functioning like a gerundive verb and qualifying the quality of the noun it modifies.

Special cases:

bʊrɑ̃sɔ (reɖɒɖeɖrɒn) rɔ\PSP sʊnd̪ɔrɔ lɔt̪ɑgʊɖɪkɔ ‘beautiful creepers of burans’

ɟɑhɑrɔ\DM_DMR mɔd̪ʰjɔ\N_NST d̪eɪ\PSP ‘through which’

3.2.28. Proper Noun-N_NNP

The proper nouns are basically some specific names which denote to one particular entity. It includes the names of person, place or thing. The examples would be rɑm, mɒhən, kɒlkɑt̪ɑ, d̪ɪlli, hɪmɑləjɔ, kɒkɑkɒlɑ etc. No separate tag has been assigned to abbreviations; they are marked as proper nouns. Acronyms, if used as proper nouns, should be marked as proper nouns and if common then as common (ILCI, 2010). No generic name has been tagged as proper nouns; nouns referring to proper persons, place or a particular thing having a proper name are tagged as proper nouns.

Names of persons, places and things:

rɑmɔ, mɒhɔnɔ, kɒlkɑt̪ɑ, d̪ɪlli, hɪmɑɭɔjɔ, kɒkɑkɒlɑ etc.

Transliterated names from other languages:

peles\N_NNP ɔn\N_NNP hʊɪlsre\N_NNP (English)

ɖɪlɔks\N_NNP selʊn\N_NNP (English)

3.2.29. Punctuation-RD_PUNC

Punctuations include the characters that are considered as the regular punctuation marks in Odia, for example- , . ? ! | - -: : ; “ ” ‘ ’ .

3.2.30. Reciprocal Pronoun-PR_PRC

In Hindi, reciprocal pronouns denote some reciprocity. This is commonly denoted by /pɔrɔspɔr/, /ɑpəs mẽ/ etc. in Hindi (ILCI, 2010).

For example pɔrɔspɔrɔ\PR_PRC mɔd̪ʰjɔre ‘with one another’

niɟɔ\PR_PRC niɟɔ\PR_PRC bʰɪt̪ɔre ‘among yourselves’

3.2.31. Reduplicative Echo Words-RD_ECH

“Reduplicated words: verbs, adverbs, classifiers, noun etc. have been tagged according to their respective categories. However, phrases like *pani-vani, chai-vai* etc contain echo words which do not belong to any POS category. In such cases, the word which belongs to a POS category should be marked with that tag and the echoword should be marked as residual echo word. Example *pani*\N_NN -\PUNC *vani*\RD_ECH” (ILCI, 2012) in Hindi. Similarly, in Odia there are certain sounds that help make a word to echo with the preceding word.

For example pɔgʰɑ\N_NN pʰɔgɑ\RD_ECH

cɑ\N_NN pʰɑ\RD_ECH

3.2.32. Reflexive Pronoun-PR_PRF

“Reflexive pronouns are the ones that denote to ownership to its antecedent which can be either a noun or a pronoun. The only examples of reflexive pronouns in Hindi are əpnɑ/əpne/əpni/, sʊɔjɔŋ, and kʰʊd̪” (ILCI, 2010). The reflexive pronouns in Odia are described below.

Examples nɪɟɔrɔ\PR_PRF sɔmɔjɔ ‘own time’

niɟe\PR_PRF krʊsnɔ ‘Krishna himself’

sʊɔjɔŋ\PR_PRF bʰɔgɔbɑnɔ ‘god himself’

sʊɔ\PR_PRF st̪ʰɑnɔkʊ ‘to own place’

ɑpɔɳɑrɔ\PR_PRF d̪ʊkʰɔsʊkʰɔkʊ ‘his/her happiness’

se sʋijɔ\PR_PRF gjɑnɔkʊ ‘his own knowledge’

3.2.33. Relative Demonstrative-DM_DMR

“The relative demonstrative occurs in the same form as the relative pronoun. The difference is only that these relatives are always followed by a noun that it modifies” (ILCI, 2010).

ɟɪs\DM_DMR gɑ̃ʋ mẽ mɛ̃ gəjɑ t̪ʰɑ ʋəh bəhʊt̪ sʊnd̪ər t̪ʰɑ “That village which I went is quite beautiful”

ɟəh\DM_DMR nəhər ʈʊʈ gəji t̪ʰɪ ʊski mərəmmət̪ ki ɟɑ rəhɪ hɛ “this bridge has been broken; it is being repaired”.

Odia examples are

ɑme sehɪ kʰet̪rɔre ɑsɪ ɟɑɪt̪ʰɑʊ ɟeʊ̃ʈʰɪ\DM_DMR ‘we get impressed by those words which’

eʈʰɑre lɔgɑɟɑɪt̪ʰɑe ɟeʊ̃t̪ʰɪre\DM_DMR ‘here it is used where’

ɟeʊ̃\DM_DMR kɑmɔre lɒkɔ ʈʰɪk hɒɪpɑrɔnt̪ɪ ‘that work by which people can be well’

bʰɔkt̪ɪre ɟe\DM_DMR kɑrɟɔ kɔrɑ hʊe ‘the work which is done out of devotion’

e git̪ɔ bʰɔkɔt̪ɔ ɟɔhĩre\DM_DMR nɑmɔ bɔɖɔmuɭɔ grɑmɔre ‘where the song and the devotee are; there the name of the village of Badamula is’

3.2.34. Relative Pronoun-PR_PRL

“The relative pronouns are those pronouns whose antecedent can be either a noun or a pronoun. However, these pronouns do not make any difference in number or gender as in the case of personal pronouns. The relative pronoun in Hindi is represented by /ɟɒ/ and its inflected forms” (ILCI, 2010).

Examples ɟɑhɑnkɔrɔ\PR_PRL sebɑkɑrɟjɔre ‘in the service of whom’

ɟɑhɑnkɔrɔ\PR_PRL sɔhɑcɑrɟjɔ\N_NN ‘whose help’

ɟe\PR_PRL mɒʈʰɑre ‘who, to me’

ɟɪe\PR_PRL sɑrirɪkɔ bjɑjɑmmɔ kɔre ‘who does physical exercise’

niɭɑmbɔrɔ ɟehʊ\PR_PRL ‘who is Nilambar’

ɟeɔ̃mɑne\PR_PRL ‘who are’

3.2.35. Subordinating Conjunctions-CC_CCS

“Subordinator conjunctions typically conjoin two clauses and the second clause is subordinated. That is the clause conjoined by the subordinator word is the subordinate clause against the main clause” (ILCI, 2010).

Coordinating words like the following:

se kɔhɪlɑ ɟe\CC_CCS t̪ɑrɔ d̪ehɔ bʰɔlɔ nɔt̪ʰɪlɑ “he told that he was not well”

ɟɔd̪ɪ\CC_CCS mʊ̃ ehɑ kɔrɪd̪ɪe t̪ebe\CC_CCS t̪ɔme kɔɳɔ d̪ebɔ “If I do it, what will you give me?”

ehɪ ʈrenɪŋ kebɔɭɔ t̪ɑɭɪ bɔɟeɪba paĩ nʊhɔ̃ bɔrɔŋ\CC_CCS peʈɔ pɒsɪbɑ sɔkɑse mɔd̪ʰjɔ ‘This training is not only for clapping but also for livelihood.”

ɟɔd̪ɪ rɔhɪbɑkʊ cɑhɔ̃ t̪ɑhɑhele\CC_CCS eʈʰɑre kicʰɪ rɪsɔrʈ mɔd̪ʰjɔ ɔcʰɪ “if you want to stay, then there are resorts here as well”

ehɑ mɔkɑʊre hĩ sɔmbʰɔbɔ ɔʈe kɑhĩkɪnɑ\CC_CCS seʈʰɑrɔ ɟɔnɔsŋkʰjɑ mɒʈɔ 5, 50, 000 “this is possible only in Macau; the population is 5.5 lakhs in total”

pʰɔɭɔ sɔɲrɔhɔ kɔrɪbɑrɔ sɔmɔjɔ set̪ebeɭe\CC_CCS hɒɪt̪ʰɑe ɟet̪ebeɭe\CC_CCS segʊɖɪkɔ pɑcɪ ɟɑɪt̪ʰɑe “the harvesting time for the fruits is then when they are ripe”

3.2.36. Unknown Words-RD_UNK

“Unknown words are the words for which a category cannot be decided by the annotator” (ILCI, 2010). These may include words and phrases or sentences from a foreign language written in Odia script (ILCI, 2010). For example, there has been a large number of Sanskrit words transliterated into Odia but incomprehensible by the annotator. These types of instances could be observed in the corpora from the literature domain.

Regular transliterated words from Sanskrit:

e̪t̪ecɑ̃sɔkɔɭɑʔ\RD_UNK pʊsɔʔ\RD_UNK krʊsnɔst̪ʊ\RD_UNK

Agglutinated transliterated words from Sanskrit:

bʰɑbɪt̪ɔmɑnɔsɔrʊsɑpʊrʊsɑbɔsɑd̪ɔsɑt̪ɔsɑt̪ɔbɔɪmʊkʰjenɔ\RD_UNK

bʰɔgbɔt̪krʊsnɔcɔɪt̪ɔnjɔmɑnɔt̪ɪrɒbʰɑbɔ\RD_UNK

3.2.37. Symbols/Special Characters-RD_SYM

The symbols are the mathematical or other special characters that are not part of the regular Odia script such as v̇, *, @, #, $, %, [,], {, }, (,), XXX etc. (ILCI, 2010).

Examples v̇\RD_SYM mrʊt̪jʊnɟɔjɔ\N_NNP rɔt̪ʰɔ\N_NNP ‘late Mrutyunjaya Rath’

*\RD_SYM d̪urɔd̪ʰɪgɔmɑrt̪ʰɔkɔ\RD_UNK srʊt̪ɪɟʊkt̪ɑbʰjɑŋ\RD_UNK

3.2.38. Verbal Noun-N_NNV

Verbal nouns in Odia do have the participial and infinitival constructions and have some of the features of the noun like definiteness, noun phrase acting like the agent and genitive case where the verbal noun is part of the possessive elements etc. (N_NNV) (Neukom, 2003).

For instance /kʰɪɑ/, /pɔhɔ̃rɑ/, /nɔcɑ/, /kʰɑɪbɑɽɑ/ etc.

Verbal nouns with a classifier marker:

t̪ɑrɔ ɑsɪbɑɽɑ\N_NNV mɒt̪e bʰɔlɔ lɑgɪlɑnɪ “I did not like his coming”.

With genitive case:

pɪlɑrɔ kɔnd̪ɑ\N_NNV ‘the boy’s crying’

ɟɔɳɔkɔrɔ lekʰɑ\N_NNV ‘one’s writing’

With an oblique case:

t̪ɑrɪɳɪcɔrɔɳɔnkɔ lekʰɑ\N_NNV ‘Tarinicarana’s writing’

rɑd̪ʰɑnɑt̪ʰɔnkɔ pɔhɔ̃rɑ\N_NNV ‘Radhanatha’s swimming’

3.3. Issues and Challenges in Annotating Odia Corpora with the BIS Annotation Scheme

There are many issues that have come to the notice during the annotation work of the Odia POS corpora. The issues pertain to the case of adverb, conjunct and serial verbs, morphophonemics, agglutination, compounds, punctuations, and affixes. They are vividly discussed below.

3.3.1. The Case of Adverb

In Odia, some of the adverbs are of single string whereas some others are consisted of more than one. BIS standard scheme addresses only the manner adverbs and mentions that adverbs of manner need to be tagged as adverbs. In most of the Indian languages, adverbs consist of one or more than one word which creates difficulty for the human annotators to annotate the manner adverbs. The reason of difficulty owes to two significant perspectives of looking at the parts of speech: the morphological approach and the syntactic approach. It has been averred that one has to go for the decision in favour of the lexical approach while annotating the parts of speech (ILCI, 2010). Thus, there is a discrepancy with regard to the decision of tagging the manner adverbs as RB and going for the morphological approach. Because, if one follows the morphological approach, most of the manner adverbs will have different improbable tags which may not treat manner as manner adverbs. In other words, they will not modify a verb. Although, this may work better for a machine learning approach to establishing the word-tag relation (ILCI, 2010) and excel the accuracy rate of the statistical tagger, but by doing so one is muting most of the linguistic information.

Examples

1. mɛ̃ jəh kɑm əccʰe se kəruŋɑ (Hindi)
2. mʊ̃ ehɪ kɑmɔ bʰɔlɔ bʰɑbɔre kɔrɪbɪ (Odia)
   I will do this work properly.
3. mɛ̃ jəh kɑm səpʰəlt̪ɑ purbək kəruŋɑ (Hindi)
4. mʊ̃ ehɪ kɑmɔ sɔpʰɔɭɔt̪ɑ purbɔkɔ kɔrɪbɪ (Odia)
   I will do this work successfully.

Morphological approach:

Examples mɛ̃\PR_PRP jəh\DM_DMD kɑm\N_NN əccʰe\JJ se\PSP kəruŋɑ\V_VM |\RD_PUNC

mɛ̃\PR_PRP jəh\DM_DMD kɑm\N_NN səpʰəlt̪ɑ\N_NN purbək\JJ kəruŋɑ\V_VM |\RD_PUNC

mʊ̃\PR_PRP ehɪ\DM_DMD kɑmɔ\N_NN bʰɔlɔ\JJ bʰɑbɔre\N_NN kɔrɪbɪ\V_VM_VF |RD_PUNC

mʊ̃\PR_PRP ehɪ\DM_DMD kɑmɔ\N_NN sɔpʰɔɭɔt̪ɑ\N_NN purbɔkɔ\JJ kɔrɪbɪ\V_VM_VF |RD_PUNC

If one adheres to the morphological approach, they are missing the syntactic feature ingrained in the sentence i.e. adverbs modify the verbs directly or indirectly irrespective of their position in the sentence; apart from modifying other grammatical categories like adjectives. In Hindi, "Postpositional phrases with nouns followed by *se* 'with' and compounds with items borrowed from Sanskrit such as /purbək/ 'with' are used as manner adverbs" (Kachru, 2006, pp.102). In example (1) Hindi, əccʰe\JJ se\PSP is the adverbial phrase which consists of an adjective followed by a preposition and in (2), səpʰəlt̪ɑ\N_NN purbək\JJ is the adverbial phrase comprising of a common noun followed by an adjective. Neukom (2003) has discussed that adjectives in Odia can be used as modifiers of verbs or of clauses. This is done in several ways:

- The adjectives appear in the same form as in attributive function.
- They take the locative case marker /-re/.
- They are combined with the converb /kɔrɪ/ 'having done'.
- They function as modifier to a head noun marked by the locative case, such as /rupɔre/ or /rupe/ 'in the form' or /bʰɑbɔ-re/ 'in the thought'.

Syntactic approach:

Examples mɛ̃\PR_PRP jəh\DM_DMD kɑm\N_NN əccʰe\RB se\RB kəruŋɑ\V_VM |\RD_PUNC

mʊ̃\PR_PRP ehɪ\DM_DMD kɑmɔ\N_NN bʰɔlɔ\RB bʰɑbɔre\RB kɔrɪbɪ\V_VM_VF |RD_PUNC

mɛ̃\PR_PRP jəh\DM_DMD kɑm\N_NN səpʰəlt̪ɑ\RB purbək\RB kəruŋɑ\V_VM |\RD_PUNC mõ\PR_PRP ehɪ\DM_DMD kɑmɔ\N_NN sɔpʰɔɭɔt̪ɑ\RB purbɔkɔ\RB kɔrɪbɪ\V_VM_VF |RD_PUNC

If one adheres to the syntactic approach, they need to annotate them all as manner adverbs. But this may create complications for the machine learing approach to handling other such cases of occurrences of these words with differently-tagged labels.

For instance,

There may be words like əccʰe\JJ kɑm\N_NN 'good work', səpʰəlt̪ɑ\N_NN in the Hindi corpora and bʰɔlɔ\JJ kɑmɔ\N_NN 'good work', sɔpʰɔɭɔt̪ɑ\N_NN being annotated with different labels from the earlier ones discussed above in the examples under morphological approach. This may create complications for the machine learning as it will face with the disambiguation issue.

3.3.2. Conjunct Verbs

"Conjunct verb is a type of complex verb in which a nominal is followed by a verb" (Majhi, 2007). Linguistically, verbs can be of two types under complex predicates: conjunct and compound verbs. Conjunct verbs have two different structures:

Examples Adjective+verb= bʰɔlɔ\JJ hʊɔ\V_VM_VF, st̪ʰɪt̪ɔ\JJ hebɑ\V_VM_VINF etc.

Noun+verb= bʰɔrɔsɑ/N_NN kɔrɔ\V_VM_VF, bɪsʊɑsɔ\N_NN kɔrɔ\V_VM_VF etc.

Transitive conjunct:

5. mɛ̃ne ʊsko nɪjɔnt̪rɪt̪\N_NN kɪjɑ\V_VM
6. mõ t̪ɑkʊ nɪjɔnt̪rɪt̪ɔ\N_NN kɔlɪ\V_VM_VF
   I controlled him.
7. mɛ̃ne ʊsko nɪmɔnt̪rɪt̪\N_NN kɪjɑ\V_VM
8. mõ t̪ɑkʊ nɪmɔnt̪rɪt̪ɔ\N_NN kɔlɪ\V_VM_VF
   I invited him.

In these examples mentioned above, if one follows the morphological approach, they have to annotate the nouns in (noun+verb) construction as adjectives; as the canonical forms of these words are adjectives in their respective languages: Hindi and Odia.

For example nɪmɔnt̪rɪt̪ɔ\JJ bjɔkt̪ɪ\N_NN

nɪjɔnt̪rɪt̪ɔ\JJ kɑrɟjɔ\N_NN

These types of cases create challenges for the human annotator whether to tag them as nouns or adjectives. Further, one can also decide in favour of annotating them as main verb which seems plausible on the part of the annotator which may thereby prove to be absolutely wrong, as the concerned words are not having any feature pertaining to PN and TAM. Therefore, when there is transitivity in a given sentence where the conjunct verb occurs, the annotator can tag them as common nouns. There are some other morpho-syntactic criteria to decide, but here it is confined to this criterion only. This is not a hard and fast rule; there could be some exceptions as at this place, an adjective can also occur.

For example

Transitive:

9. mɛ̃ne ʊsko ʈʰɪk\JJ kɪjɑ\V_VM
10. mʊ̃ t̪ɑkʊ ʈʰɪk\JJ kɔlɪ\V_VM_VF

Intransitive/existential conjunct:

11. mɛ̃ ken̪d̪rit̪\JJ hʊ̃\V_VM
12. mʊ̃ ken̪d̪rit̪ɔ\JJ ɔʈe\V_VM_VF
    I am focussed.
    or I am controlled.

In these instances drawn above, it can be stated that when there is an intransitive sentence, the nouns in (noun+verb) function as adjectives because one can replace the word with the other adjectives.

3.3.3. Serial Verbs

When a series of verbs occur sequentially, they are known as serial verbs.

13. se mɑrkɔrɪ d̪ɔʊɖɪ cɑlɪgɔlɑ
    ‘Beating me he went’.

In this example, it is quite ambiguous whether to tag /d̪ɔʊɖɪ/ as the main verb or the non-finite occurrence of the verb like the preceding non-finite verb /mɑrɪkɔrɪ/. The ambiguity owes to the fact that in Odia, both /-ɪ/ and /kɔrɪ/ are used as non-finite markers or the conjunctive participle markers indicating non-finiteness. So, the second reading for the same sentence will be as follows:

14. *se mɑrɪkɔrɪ ɖ̪ɔʊɖɪkɔrɪ cɑlɪɡɔlɑ
    he went running beating me.

If two non-finite verbs are occurring simultaneously, they have to be conjoined by a co-ordination like the following sentence.

15. se mɑrɪkɔrɪ o ɖ̪ɔʊɖɪkɔrɪ cɑlɪɡɔlɑ.
    He went running by beating me.

If one makes the first verb as the main verb and the second as the non-finite, then the sentence can be acceptable as in the following. Another acceptable sentence could be to make the second occurrence as the main verb by making the first as the non-finite.

16. se\PR_PRP mɑrɪ\V_VM ɖ̪ɔʊɖɪkɔrɪ\V_VM_VNF cɑlɪɡɔlɑ\V_VM_VF
17. se\PR_PRP mɑrɪkɔrɪ\V_VM_VNF ɖ̪ɔʊɖɪ\V_VM cɑlɪɡɔlɑ\V_VM_VF

At often times, it is noticed that same sentence has different verbs if the writing convention is different.

For example,

18. se\PR_PRP mɑrɪ\V_VM kɔrɪ\V_VM_VNF ɖ̪ɔʊɖɪ\V_VM cɑlɪɡɔlɑ\V_VM_VF
19. se\PR_PRP mɑrɪ\V_VM kɔrɪ\V_VAUX ɖ̪ɔʊɖɪ\V_VM cɑlɪɡɔlɑ\V_VM_VF

In the above examples (16) (17), whether to annotate the verb kɔrɪ\V_VM_VNF as nonfinite or kɔrɪ\V_VAUX as auxiliary is an ambiguous and problematic case to agree upon among the human annotators. This fact will be pretty clear if one looks at the detailed chart of the Inter Annotator agreement[57] and the related appendix.

3.3.4. Sandhi Phenomenon or Morphophonemics

Sandhis have sound alternation within words at the morphophonemic level at the morphemic boundaries. They are broadly classified into three types:

a. Vowel Sandhi

[57] For an overall result of the IA agreement report, please refer to the section 4.2.2.1.

b. Consonant Sandhi

c. Visarga Sandhi

- **Vowel Sandhi:**

Examples sɔrbɔ\PR_PRI + ɔd̪ʰɪk\QT_QTF = sɔrbɑd̪ʰɪk\QT_QTF ‘most of all’

sɔrbɔ\PR_PRI + ʊt̪krʊsʈ\JJ = sɔrbɒt̪krʊsʈ\JJ ‘best of all’

In the examples drawn above, it is revealed by the fact that /sɔrbɔ/ is an indefinite pronoun and /ɔd̪ʰɪkɔ/ is a general quantifier. After the morphophonemic alternation of the sound where /ɔ/+/ɔ/ becomes /ɑ/, /sɔrbɑd̪ʰɪk/ becomes a general quantifier. Similar is the case with the next example where sɔrbɔ\PR_PRI + ʊt̪krʊsʈ\JJ get combined to make sɔrbɒt̪krʊsʈ\JJ.

- **Consonant Sandhi:**

Examples sɔd̪\JJ + pɔt̪ʰɔ\N_NN = sɔt̪pɔt̪ʰɔ\N_NN ‘good path’

sɔrɔd̪\N_NN + kɑɭinɔ\JJ = sɔrɔt̪kɑɭinɔ\JJ ‘autumnal’

The instances drawn above demonstrate the fact that /sɔd̪/ is the adjective and /pɔt̪ʰɔ/ is a common noun. When both these words get combined, after the alternation, the output becomes a noun sɔt̪pɔt̪ʰɔ\N_NN. Analogously, sɔrɔd̪\N_NN + kɑɭinɔ\JJ get combined to make sɔrɔt̪kɑɭinɔ\JJ which is tagged as adjective.

- **Visarga Sandhi:**

Generally, these types of words having visarga sandhi are loaned from the Sanskrit language. The morphoponemic alternations occur primarily with the sounds like visarga changing to /r/ and /s/.

Examples ɔnt̪ɔʔ\JJ + d̪ʊɔnd̪ɔ\N_NN = ɔnt̪ɔrd̪ʊɔnd̪ɔ\N_NN ‘internal confusion’

ɔnt̪ɔʔ\JJ + d̪ʊɔnd̪ijɔ\JJ = ɔnt̪ɔrd̪ʊɔnd̪ijɔ\JJ ‘internal confusion-like’

cɔt̪ʊʔ\JJ + pɔdɔ\N_NN = cɔt̪ʊspɔdɔ\N_NN ‘quadruped’

cɔt̪ʊʔ\JJ + pɔdijɔ\JJ = cɔt̪ʊspɔdijɔ\JJ ‘quadruped-like’

In these above-mentioned examples, it is obvious that /ɔnt̪ɔʔ/, which is an adjective, gets attched with /d̪ʊɔnd̪ɔ/ which is a common noun to become ɔnt̪ɔrd̪ʊɔnd̪ɔ creating again a common noun. Similar is the case with the word /cɔt̪ʊspɔd̪ɔ/. On the other hand, when an adjective gets attached with another adjective it becomes an adjective. For example, /ɔnt̪ɔʔ/ is getting combined with /d̪ʊɔnd̪ijɔ/ to make ɔnt̪ɔrd̪ʊɔnd̪ijɔ which is an adjective. Similar is the case with the word cɔt̪ʊspɔdijɔ which is an adjective.

Since both the divided parts of all the words fall in any of the categories specified under BIS standard annotation scheme, it has been agreed upon the decision to annotate these words considering the tag of the head word or the headedness feature of Odia. Mohapatra (2010) as cited in (Jena et al, 2011) has stated that "Odia is syntactically a head-final language" and therefore, it has been decided that one needs to tag the word based on the tag of the final word in a sandhi.

3.3.5. Handling Agglutination

"Agglutinative language is a language in which words are made of a linear sequence of distinct morphemes and each component of meaning is represented by its own morpheme" (SIL International, 2004).

Mohapatra (2010), as cited in (Jena et al, 2011), has also averred the fact that Odia is not only "syntactically a head-final language" but also "morphologically an agglutinating language." Furthermore, in Odia, "the suffixes, postpositions, and case endings agglutinate with the verbs, nouns, adverbs or pronouns: also one or more suffixes can combine with the base word" (Padhy and Mohanty, 2013).

3.3.5.1.Agglutination in Nouns

a) Case markers:

Examples lɒkɔ-rɔ 'man's' (genitive case marker)

lɒkɔ-kʊ 'to the man' (dative case marker)

lɒkɔ-nkʊ 'to the men' (oblique-dative case marker)

lɒkɔ-nkɔ/nkɔrɔ 'of the men' (oblique-genitive case marker)

lɒkɔ-rʊ 'from the man' (ablative case marker)

lɒkɔ-ʈʰɑre 'at the man' (locative case marker)

cʰʊrɪ-sɔhɔ 'with the knife' (instrumental case marker)

b) Nominal suffixes for number:

Singular number suffixes are /ʈɑ/, /ʈɪ/, and /ʈɪe/

lɒkɔ-ʈɑ/ʈɪ/ʈɪe ‘the man’ (singular definite and indefinite classifier markers)

The plural number suffixes in Odia are e, mɑne, mɑnɔ, gʊɖɪkɔ, gʊɖɪe, sɔmuhɔ, sreɳi, bɔrgɔ, d̪ɔɭɔ, sɔbʊ, and mɑɭɔ

Examples bɑɭɔkɔ + e = bɑɭɔke ‘boys’

bɑɭɔkɔ + mɑne = bɑɭɔkɔmɑne ‘boys’

gɔcʰɔ + gʊɖɪkɔ = gɔcʰɔgʊɖɪkɔ ‘trees’

d̪ʊrgɔ + sɔmuhɔ = d̪ʊrgɔsɔmuhɔ ‘forts’

net̪rɔ + bɔrgɔ = net̪rɔbɔrgɔ ‘eyes’

pɔrbɔt̪ɔ + mɑɭɑ = pɔrbɔt̪ɔmɑɭɑ ‘mountain ranges’

sɔhɔrɔ + mɑnɔ = sɔhɔrɔmɑnɔ ‘cities’

pɔrbɔt̪ɔ + sreɳi = pɔrbɔt̪ɔsreɳi ‘mountain ranges’

kɔpɒt̪ɔ + d̪ɔɭɔ = kɔpɒt̪ɔd̪ɔɭɔ ‘pigeons’

gʰɔrɔ + sɔbʊ = gʰɔrɔsɔbʊ ‘houses’

3.3.5.2.Agglutination in Verbs

In the non-finite forms of the verbs, the same case markers used for nouns are used, but they function in different ways performing various functions when attached with the verbs. They are as follows.

For instance

nebɑ-rɔ ‘of taking’ (the genitive marker for the non-finite verb)

nebɑ-rʊ ‘because of taking’ (the locative/ablative marker for the non-finite verb used as causal)

nebɑ-re ‘in taking’ (the locative marker for the non-finite verb)

nebɑ-ʈɑ ‘taking’ (the classifier marker used for verbal noun) nebɑ-kʊ ‘to take’ (to + infinitive marker used for an infinitive verb)

Besides, the verbal inflections for the PN and the TAM features also agglutinate with the verbs as each morpheme attached to these forms has a specific corresponding meaning.

root	**p/n**	**TENSE**											
		PRESENT				**PAST**				**FUTURE**			
		INDF	PROG	PFV	PFV.PROG	INDF	PROG	PFV	PFV.PROG	INDF	PROG	PFV	PFV.PROG
kʰɑ	1.sg	e	ʊ-ɔcʰ-ɪ	ɪ-ɔcʰ-ɪ	ɪ-ɑs-ʊ-ɔcʰ-ɪ	ɪl-ɪ	ʊ-t̪ʰɪl-ɪ	ɪ-t̪ʰɪl-ɪ	ɪ-ɑs-ʊ-t̪ʰɪl-ɪ	ɪb-ɪ	ʊ-t̪ʰɪb-ɪ	ɪ-t̪ʰɪb-ɪ	ɪ-ɑs-ʊ-t̪ʰɪb-ɪ
kʰɑ	1.pl	ʊ	ʊ-ɔcʰ-ʊ	ɪ-ɔcʰ-ʊ	ɪ-ɑs-ʊ-ɔcʰ-ʊ	ɪl-ʊ	ʊ-t̪ʰɪl-ʊ	ɪ-t̪ʰɪl-ʊ	ɪ-ɑs-ʊ-t̪ʰɪl-ʊ	ɪb-ʊ	ʊ-t̪ʰɪb-ʊ	ɪ-t̪ʰɪb-ʊ	ɪ-ɑs-ʊ-t̪ʰɪb-ʊ
kʰɑ (-hon)	2.sg	ɔ	ʊ-ɔcʰ-ɔ	ɪ-ɔcʰ-ɔ	ɪ-ɑs-ʊ-ɔcʰ-ɔ	ɪlɔ	ʊ-t̪ʰɪl-ɔ	ɪ-t̪ʰɪl-ɔ	ɪ-ɑs-ʊ-t̪ʰɪl-ɔ	ɪb-ɔ	ʊ-t̪ʰɪb-ɔ	ɪ-t̪ʰɪb-ɔ	ɪ-ɑs-ʊ-t̪ʰɪb-ɔ
kʰɑ (-hon, /-ɪ/nformal)	2.sg	ʊ	ʊ-ɔcʰ-ʊ	ɪ-ɔcʰ-ʊ	ɪ-ɑs-ʊ-ɔcʰ-ʊ	ɪl-ʊ	ʊ-t̪ʰɪl-ʊ	ɪ-t̪ʰɪl-ʊ	ɪ-ɑs-ʊ-t̪ʰɪl-ʊ	ɪb-ʊ	ʊ-t̪ʰɪb-ʊ	ɪ-t̪ʰɪb-ʊ	ɪ-ɑs-ʊ-t̪ʰɪb-ʊ
kʰɑ (+hon)	2.sg	ɑ-nt̪ɪ	ʊ-ɔcʰ-ɔnt̪ɪ	ɪ-ɔcʰ-ɔnt̪ɪ	ɪ-ɑs-ʊ-ɔcʰ-ɔnt̪ɪ	ɪl-e	ʊ-t̪ʰɪl-e	ɪ-t̪ʰɪl-e	ɪ-ɑs-ʊ-t̪ʰɪl-e	ɪb-e	ʊ-t̪ʰɪb-e	ɪ-t̪ʰɪb-e	ɪ-ɑs-ʊ-t̪ʰɪb-e
kʰɑ (+hon)	2.pl	ɑ-nt̪ɪ	ʊ-ɔcʰ-ɔnt̪ɪ	ɪ-ɔcʰ-ɔnt̪ɪ	ɪ-ɑs-ʊ-ɔcʰ-ɔnt̪ɪ	ɪl-e	ʊ-t̪ʰɪl-e	ɪ-t̪ʰɪl-e	ɪ-ɑs-ʊ-t̪ʰɪl-e	ɪb-e	ʊ-t̪ʰɪb-e	ɪ-t̪ʰɪb-e	ɪ-ɑs-ʊ-t̪ʰɪb-e
kʰɑ (-hon)	3.sg	e	ʊ-ɔcʰ-ɪ	ɪ-ɔcʰ-ɪ	ɪ-ɑs-ʊ-ɔcʰ-ɪ	ɪl-ɑ	ʊ-t̪ʰɪl-ɑ	ɪ-t̪ʰɪl-ɑ	ɪ-ɑs-ʊ-t̪ʰɪl-ɑ	ɪb-ɔ	ʊ-t̪ʰɪb-ɔ	ɪ-t̪ʰɪb-ɔ	ɪ-ɑs-ʊ-t̪ʰɪb-ɔ
kʰɑ (+hon)	3.sg	ɑ-nt̪ɪ	ʊ-ɔcʰ-ɔnt̪ɪ	ɪ-ɔcʰ-ɔnt̪ɪ	ɪ-ɑs-ʊ-ɔcʰ-ɔnt̪ɪ	ɪl-e	ʊ-t̪ʰɪl-e	ɪ-t̪ʰɪl-e	ɪ-ɑs-ʊ-t̪ʰɪl-e	ɪb-e	ʊ-t̪ʰɪb-e	ɪ-t̪ʰɪb-e	ɪ-ɑs-ʊ-t̪ʰɪb-e

kʰɑ	3. pl	ɑ-nt̪ɪ	ʊ-ɔcʰ-ɔnt̪ɪ	ɪ-ɔcʰ-ɔnt̪ɪ	ɪ-ɑs-ʊ-ɔcʰ-ɔnt̪ɪ	ɪl-e	ʊ-t̪ʰɪl-e	ɪ-t̪ʰɪl-e	ɪ-ɑs-ʊ-t̪ʰɪl-e	ɪb-e	ʊ-t̪ʰɪb-e	ɪ-t̪ʰɪb-e	ɪ-ɑs-ʊ-t̪ʰɪb-e
kʰɑ (non-human)	3. pl	e	ʊ-ɔcʰ-ɪ	ɪ-ɔcʰ-ɪ	ɪ-ɑs-ʊ-ɔcʰ-ɪ	ɪl-ɑ	ʊ-t̪ʰɪl-ɑ	ɪ-t̪ʰɪl-ɑ	ɪ-ɑs-ʊ-t̪ʰɪl-ɑ	ɪb-ɔ	ʊ-t̪ʰɪb-ɔ	ɪ-t̪ʰɪb-ɔ	ɪ-ɑs-ʊ-t̪ʰɪb-ɔ

Table. 5. Inflection and Agglutination

All the verbs inflecting for either of the TAM features have been tagged as finite verbs and if not, they have been tagged depending upon the context they are in. In the natural language data, there could be some frequent cases of unevenness so far as the consistency of the data is concerned. In the ILCI corpora, one can find much inconsistent data. If there is inconsistency, the data has been annotated according to the suitability based on the context. For instance, in the following examples one can find an excerpt of uneven data structures with regard to the verbs.

Conventional	**Unconventional**
kʰɑɪɔcʰɪ\V_VM_VF	kʰɑɪ\V_VM ɔcʰɪ\V_VM_VF
kʰɑɪsɑrɪɔcʰɪ\V_VM_VF	kʰɑɪ\V_VM sɑrɪ\V_VM_VNF ɔcʰɪ\V_VM_VF
kʰɑɪkɔrɪ\V_VM_VNF ɑsʊɔcʰɪ\V_VM_VF	kʰɑɪ\V_VM kɔrɪ\V_VM_VNF ɑsʊɔcʰɪ\V_VM_VF
pɔɖʰʊPɔɖʰʊ\V_VM_VNF	pɔɖʰʊ\V_VM_VNF pɔɖʰʊ\V_VM_VNF
nɑcɔkɔrɪ\V_VM_VNF sɒɪpɔɖɪlɑ\V_VM_VF	nɑcɔkɔrɪ\V_VM_VNF sɒɪ\V_VM pɔɖɪlɑ\V_VM_VF
bʰɑt̪ɔ\N_NN kʰɑɪ\V_VM_VNF	bʰɑt̪ɔkʰɑɪ\V_VM_VNF
kɔrɪt̪ʰɪbɑ\V_VM_VNG	kɔrɪ\V_VM t̪ʰɪbɑ\V_VM_VNG
kɔrɪ-nɔ-t̪ʰɪbɑ\V_VM_VNG	kɔrɪ\V_VM nɔ\RP_NEG t̪ʰɪbɑ\V_VM_VNG

Table. 6. Conventional and Unconventional Orthography of Odia Verbs

When the data is conventionally written in a standard form, the verbs agglutinate with respect to the PN and TAM features.

3.3.5.3.Agglutination in Temporal and Spatial Adverbs or Nouns

Generally, locative, genitive and ablative case markers get attached with the adverbs of location and time in Odia.

a) Locative case:

Examples e-ʈʰɑre ‘at this place’

se-ʈʰɑre ‘at that place’

ʊpɔrɔ-re or ʊpɔre ‘on the above’

t̪ɔɭɔ-re or t̪ɔɭe ‘under the below’

purbɔ-re or purbe ‘before’

pɔre ‘after’

b) Genitive case:

Examples eʈʰɑ-kɑrɔ ‘of this place’

seʈʰɑ-kɑrɔ ‘of that place’

ʊpɔrɔ-rɔ ‘of the above’

t̪ɔɭɔ-rɔ ‘of the under’

purbɔ-rɔ ‘of the before’

pɔrɔ-rɔ ‘of the after’

c) Ablative case:

Examples e-ʈʰɑrʊ ‘from this place’

se-ʈʰɑrʊ ‘from that place’

ʊpɔrɔ-rʊ ‘from the above’

t̪ɔɭɔ-rʊ ‘from the below’

purbɔ-rʊ or purbe ‘from before’

pɔrɔ-rʊ ‘from after’

3.3.5.4.Agglutination in Pronouns and Demonstratives

Pronouns	Person	Number	Genitive case	Ablative case	Dative case
mʊ̃	first	singular	mɒ-rɔ	mɒ-ʈʰɑrʊ	mɒ-t̪e
t̪ʊ (-honorific)	second	singular	t̪ɒ-rɔ	t̪ɒ-ʈʰɑrʊ	t̪ɔ-t̪e

t̪ʊme (-honorific)	second	singular	t̪ʊmɔ-rɔ	t̪ʊmɔ-ʈʰɑrʊ	t̪ʊmɔ-kʊ
ɑpɔɳɔ (+honorific)	second	Singular	ɑpɔɳɔnkɔ-rɔ	ɑpɔɳɔnkɔ-ʈʰɑrʊ	ɑpɔɳɔ-nkʊ
ɑme or ɑmbʰe	second	Plural	ɑmɔ-rɔ	ɑmɔ-ʈʰɑrʊ	ɑmɔ-kʊ
Se	third	Singular	t̪ɑ-rɔ/t̪ɑnkɔ-rɔ	t̪ɑ-rɔ/t̪ɑnkɔ-ʈʰɑrʊ	t̪ɑ-kʊ
semɑne	third	plural	semɑnɔnkɔ-rɔ	semɑnɔnkɔ-ʈʰɑrʊ	semɑn-nkʊ
Demonstratives	**Person**	**Number**	**Genitive case**	**Ablative case**	**Dative case**
ehɑ		singular	ehɑ-rɔ	ehɑ-ʈʰɑrʊ	ehɑ-kʊ
t̪ɑhɑ		singular	t̪ɑhɑ-rɔ	t̪ɑhɑ-ʈʰɑrʊ	t̪ɑhɑ-kʊ
egʊɖɪkɔ		plural	egʊɖɪkɔ-rɔ	egʊɖɪkɔ-ʈʰɑrʊ	egʊɖɪk-kʊ
segʊɖɪkɔ		plural	segʊɖɪkɔ-rɔ	segʊɖɪkɔ-ʈʰɑrʊ	segʊɖɪk-kʊ
eɪ		singular	eɪʈɑ-rɔ	eɪ-ʈʰɑrʊ	eɪʈɑ-kʊ
seɪ		singular	seɪʈɑ-rɔ	seɪ-ʈʰɑrʊ	seɪʈɑ-kʊ

Table. 7. Agglutination in Pronouns and Demonstratives

Categories having classifier markers have been decided for annotating as classifiers RP_CL apart from the verbs. When postpositions get attached to different categories, there is no any significant problem. So, in this case, it has been agreed upon that these words have to be annotated based on the labels of their respective categories except the verbs as infinitive marker for verbs and dative marker for nouns are the same. Case endings also do not affect the categories of the verbs. At often times, postpositions and case endings are used as follows in the natural language corpora.

Examples (\RD_SYM rɑmɔ\N_NNP ɒ\CC_CCD hɔrɪ\N_NNP)\RD_SYM rɔ\PSP

(\RD_SYM bʰɑrɔt̪ɔrɔ\N_NNP prad̪ʰɑnɔmɔnt̪ri\N_NN ɒ\CC_CCD cinrɔ\N_NNP rɑsʈrɔpɔt̪ɪ\N_NN)\RD_SYM nkʊ\PSP

Instances like the above pose significant problems as the case markers are attached unconventionally after the symbol right round bracket.

3.3.6. Dealing with Compounds

All the compounds attached with a punctuation have been separated and tagged according to their respective categories as per the BIS tagset. In this section one is especially dealing not with the types of compunds like copulative, determinative, attributive, adverbial, numeral appositional and appositional (Majhi, 2007) in Odia, but with the compounds made with the help of hyphens and other such punctuations. The following discussions are under three heads: nominal, collocative and reduplicative compounds.

3.3.6.1.Nominal Compounds

In the examples,

ɟɔn-ɪn-d̪ɪ-ʋɪldɔrnes\N_NNP memɒrɪɑl cɔrc

John/-ɪ/n-the-Wilderness Memorial Church

The English-derived compound expresion ‘John/-ɪ/n-the-Wilderness’ is attached with the hyphens and will be misleading, if one tags the whole expression as proper noun; even though by separating or conjoining. If one disintegrates the whole compound, the tags will differ.

Example ɟɔn\N_NNP -\RD_PUNC ɪn\PSP -\RD_PUNC d̪ɪ\N_NN -\RD_PUNC ʋɪldɔrnes\N_NN

In this way, one will end up in misinformation and by the way, the tagset is not meant for English, an Eurpopean language; at least, not at the parts of speech level. Therefore, it is imperative that one needs to make them one compound and tag them as per their category. Similarly, if they are already separated in the corpora itself, one needs to make them a single entity and tag them.

Example nɪrlɔɟɔ\JJ pɔɳɪɑ\N_NN ‘shameless-ness’

nɪrlɔɟɔpɔɳɪɑ\N_NN ‘shamelessness’

The above-mentioned word is a common noun as a whole, but in isolation, the first entity is an adjective while the second is a common noun. While deciding the tag of these types of words, the fact of right-headedness has been considered. Hence, since the head of the compound is a common noun, the tag goes in favour of it.

Similarly, the word /sɔrbɔ-sɔmmɔt̪ɪ/ is a common noun as a single entity. If one separates the two elements, they find out that the first element is indefinite pronoun and the second is a common noun.

Examples sɔrbɔ-sɔmmɔt̪ɪ\N_NN ‘all-consent’

sɔrbɔ\PR_PRI sɔmmɔt̪ɪ\N_NN ‘all consent’

At often times, the unconventionally-written words pose significant problem for annotation.

Example the phrase sɔpʰɔɭɔt̪ɑpurbɔkɔ\RB ‘successfully’ is an adverbial phrase

One will tend to tag them as common noun and adjective because they have the features of the respective categories.

Example sɔpʰɔɭɔt̪ɑ\N_NN -\RD_PUNC purbɔkɔ\JJ ‘successfully’

3.3.6.2.Words in Collocation

If one separates some of the collocative compounds, there is no significant problem. But some others create issues when isolated.

For example bʰɔlɔmɔnd̪ɔ\N_NN kʰɑɔ\V_VM_VF “eat good-bad things”

bʰɔlɔ\JJ mɔnd̪ɔ\JJ kʰɑɔ\V_VM_VF “eat good-bad”

In the first example, the compound functions like a common noun as there is no any head that the adjectives modify. But when the compound is separated, the elements behave like adjectives even though in absentia of the common noun as the head. The idea will become more obvious if one takes the following example.

Example se\PR_PRP bʰɔlɔmɔnd̪ɔ\N_NN bʊɟʰenɪ\V_VM_VF

Here the two components in compound behave like a common noun. When they are used with the addition of one head noun, they seem to be adjectives even after separated like the following.

Example se\PR_PRP bʰɔlɔ\JJ mɔnd̪ɔ\JJ Kɔt̪hɑ\N_NN bʊɟʰenɪ\V_VM_VF

The words in collocation that do not create problem even after separated are as follows.

Example bɔhɪ-pɔt̪rɔ, ʈɔnkɑ-pɔɪsɑ, kɔt̪ʰɑ-bɑrt̪ɑ, nɑcɔ-git̪ɔ, khɑɪbɑ-pɪɪbɑ, sɔpʰɑ-sʊt̪ʊrɑ, lekʰɪbɑ-pɔɖʰɪbɑetc.

3.3.6.3.Reduplicative Expressions

"It is the repetition of a segment, a syllable, or some part or whole of a lexical or phrasal unit leading to a semantic or grammatical modification" (Pandey, 2007) as cited in (Majhi, 2007). "In such formation, the derived word is constituted of two elements: the base form and the reduplicant" (Majhi, 2007). There are two types of reduplication: partial and total. In total reduplication, the whole part of the base is reduplicated and in the partial reduplication, some part is reduplicated.

- **Total Reduplication:**

Non-finite verbs:

Examples Progressive: kʰeɭʊ-kʰeɭʊ, ɟɑʊ-ɟɑʊ, cɑhʊ̃-cɑhʊ̃ etc.

Perfective: kʰeɭɪ-kʰeɭɪ, ɟɑɪ-ɟɑɪ, cɑhɪ̃-cɑhɪ̃ etc.

Adverbial:

Examples d̪ʰɪre-d̪ʰɪre, bʰɔlɔre-bʰɔlɔre, sigʰrɔ-sigʰrɔ etc.

Adjective:

Examples ʈʰɪk-ʈʰɪk, bʰɔlɔ-bʰɔlɔ, kʰɔrɑpɔ-kʰɔrɑpɔ etc.

Sub-ordinating conjunction:

Examples ɟemɪt̪ɪ-ɟemɪt̪ɪ, semɪt̪ɪ-semɪt̪ɪ etc.

Onomatopoeic:

Examples sɑɪ̃-sɑɪ̃, ʈɪk-ʈɪk, pʰɔr-pʰɔr, ɟʰɔr-ɟʰɔr, kʰɔr-kʰɔr, t̪ʰɔr-t̪ʰɔr etc.

- **Partial Reduplication:**

Verbal noun:

Examples kʰɪɑ-kʰʊɪ, pɔɖʰɑ-pɔɖʰɪ, kʰeɭɑ-kʰeɭɪ, nɔcɑ-nɔcɪ etc.

Adjective:

Examples bʰɔlɔ-bʰelɔ, ʈʰɪkɔ-ʈʰɑkɔ etc.

As per the BIS standard annotation scheme, there is no category specified for the reduplication phenomenon which is one of the important linguistic phenomena in many of the Indian languages (Abbi, 2001). If one is separating the elements in a reduplicated expression and marking as per their categories, they are not justifying with the words and are missing out information that are vital for any language.

3.3.7. Punctuations

Punctuations in Odia have variegated functions other than their canonical functions. They can be used for punctuating, co-ordinating, marking list items, compounding, as section headers, and joining frozen expressions. Marking all punctuations as RD_PUNC ‘may be misleading’ as put forth by (Edna et al., 2012). But the punctuations like closing inverted comma and hyphenation function differently in different contexts. They are vividly dealt with in the following sections.

3.3.7.1.Hyphenated Expressions

- Hyphenation is used for almost all types of compounding like collocative, reduplicative, echo-words and so on.

Examples bɔhɪ-pɔt̪rɔ (collocation)

kʰeɭɑ-kʰeɭɪ (reduplication)

cɑ’-pʰɑ (ଚା’ଫା) (Echo-word formation)

(cɑ\N_NN ’\RD_PUNC -\RD_PUNC pʰɑ\N_NN)

In all these cases, marking all the punctuations as punctuations (RD_PUNC) may mute much linguistic information ingrained in a given language. One can notice the fact that the inverted comma used in the echo-word is not a canonical comma of inverted expression. There are some other specific words where the inverted comma is used not in Odia as enclosing some part of the sentence; which is one of its canonical functions.

- **List Item Marker:**

Hyphenation is also used as a list item marker for separating items in a list.

Example mɒpaĩ\PR_PRP bɔɟarɔrʊ\N_NN ɟɪnɪsɔgʊɖɪkɔ\N_NN ɑɳɪd̪ebɔ\V_VM_VF -\RD_PUNC kʰeɭɔɳɑ\N_NN ,\RD_PUNC mɪʈʰɑ\N_NN ,\RD_PUNC ɒ\CC_CCD cɪnɑbad̪ɑm\N_NN |\RD_PUNC

- **Section Header:**

It is further used as section headers. These sorts of examples could be noticed in the domains of tourism and especially in the descriptive part of a particular location, person, event etc.

Example nɔksɑpɑlı\N_NNP :\RD_PUNC or -\RD_PUNC ehɑ\DM_DMD sɔmbɔlpʊrst̪ʰɪt̪ɔ\JJ ekɔ\QT_QTC cʰɒʈɪɑ\JJ gɑ̃\N_NN

- **Co-ordination:**

Hyphenation can further be used as the co-ordinating conjunction. It can conjoin two words, phrases and clauses of equal linguistic status.

Examples (rɑmɔ-hɔrı)rɔ gɑı\N_NN

rɑmɔrɔ gɑı-hɔrırɔ gɑı

- **Subordination:**

In the below-stated example, it can be found out that hyphenation functions like a subordinating conjunction in the form of a ‘that’ complement.

For example lɒke\N_NN kɔhɔnt̪ɪ\V_VM_VF -\RD_PUNC bɪɟɔjɔ\N_NNP kʊɑɖe\PR_PRQ ɟɔɳe\RP_CL bʰɔlɔ\JJ lɒkɔ\N_NN

3.3.7.2.Inverted Comma

Canonically, inverted commas function as to enclose a reported speech, for emphasis, to quote already averred statements etc.

For example se kɔhılɑ, “\RD_PUNC mɔ̃ kɑlı ɟɑıpɑrıbı nɑhĩ ”\RD_PUNC

However, there are some other cases where single inverted commas are used in Odia text not to indicate their canonical function, but something unique.

Examples t̪ɑ’rɔ\PR_PRP

cɑ’\N_NN

k’ɔɳɔ\DM_DMQ

If one breaks them as special character or inverted comma, then one will face the issue of the loss of linguistic information.

Examples t̪ɑ\PR_PRP ’\RD_PUNC rɔ\PSP

cɑ\N_NN ’\RD_PUNC

k\N_NN ’\RD_PUNC ɔɳɔ\N_NN

These cases are pointed out, because when one tokenizes the data with an automatic tokenizer, one faces them.

3.3.7.3.Colon

Colons can also be used as separating the list items and co-ordinating, apart from their canonical functions.

- **Separating Items as a List Item Marker:**

Example bɔhʊ\QT_QTF prɔkarɔrɔ\N_NN rɔsɔ\N_NN ɔcʰɪ\V_VM_VF :\RD_PUNC somɔ\N_NN rɔsɔ\N_NN ,\RD_PUNC srʊngarɔ\N_NN rɔsɔ\N_NN ,\RD_PUNC ɪt̪jad̪ɪ\QT_QTF

- **As a Co-ordinator:**

Example seʈʰarʊ\N_NST ɪnɖɪan\N_NNP mɔharaɟarɔ\N_NNP d̪ʊɪʈɪ\RP_CL ʈrɪp\N_NN ɔcʰɪ\V_VM_VF :\RD_PUNC gɒʈɪe\RP_CL mʊmbaɪrʊ\N_NNP d̪ɪlli\N_NNP ɒ\CC_CCD ɔnjɔʈɪ\RP_CL d̪ɪllirʊ\N_NNP kɒlkat̪a\N_NNP

3.3.8. Handling Prefixes and Suffixes

The following prefixes and suffixes are not to be separated from the root form of the word to which they are attached; whether they are prefixed or suffixed with or without hyphens.

- **Untokenized Prefixes:**

UNTOKENIZED PREFIXES					
ɔ-	sʊ-	ɪ-	d̪ɔrɔ-	cɔt̪ʊʔ-	pɔrɪ-
kʊ-	ɔɳɔ-	i-	ɔd̪ʰɔ-	sɔd̪-	ɔbɪ-
ʊpɔ-	ɔbɔ-	ɪt̪ɔʔ-	ɔɳɔ-	ɔnɔ-	ɔbɔ-
nari-	a-	ʊt̪-	ɔt̪ɪ-	ɔnʊ-	mɔnd̪ɔ-
mɔhɪɭa-	d̪ʊʔ-	ʊd̪-	ɔnt̪ɔʔ-	prɔt̪ɪ-	
sɔrbɔ-	kʰɔnɖe-	bɔhʊ-	pɔrɪ-	sɔmɔ-	

Table. 8. Untokenized Prefixes in Odia

- **Untokenized Suffixes**:

So far as suffixes are concerned, they are of three types mentioned below: primary derived nominal suffixes, secondary derived nominal suffixes, and compounds. The suffixes can be used to derive adjectives, verbs (tense, aspect etc.) and so on from nouns. Verbal derivational suffixes are used to make nouns or verbal nouns. Further, noun-noun derivational suffixes like (plurals, case markers, classifiers etc.) have also been provided in the table below. “A morph /mɑne/ is used to indicate plurality form only for human. As stated by G.A. Grierson and Dr. S.K. Chatterjee, It is unique in Odia“ (Pattanayak and Prushty, 2013).

Noun>noun

Examples bɑɭɔkɔ+mɑne= bɑɭɔkɔmɑne

gɔcʰɔ+kʊ= gɔcʰɔkʊ

lɒkɔ+ʈɪ= lɒkɔʈɪ

Noun>adjective

Examples bɔɭɔ+bɔnt̪ɔ= bɔɭɔbɔnt̪ɔ

d̪ɑnt̪ɔ+ʊrɑ= d̪ɑnt̪ɔ+ʊrɑ

Noun>verb

Examples rɔŋɔ+ɪbɑ= rɔŋeɪbɑ

kɔrɔt̪ɔ+ɪbɑ= kɔrɔt̪ɪbɑ

Adj>verb

Examples ɒsɑrɔ+ɪbɑ= ɒsɑrɪbɑ

sɔɭɔkʰɔ+ɪbɑ= sɔɭɔkʰɪbɑ

melɑ+ɪbɑ= melɑɪbɑ

Verb>noun

Examples lekʰ+ɑ= lekʰɑ

pɔ̃hɔ̃r+ɑ= pɔ̃hɔ̃rɑ

UNTOKENIZED SUFFIXES							
-kant̪ɔ	/-ɪ/	- ʈʰare	-ʊa	-ɪarɔ	/-ɪ/ɳɔ	-ka	-ʈɪa
-kɭɪsʈɔ	-mane	- manɔnkɔrɔ	-aɭɪ	-ɪkɔ	/-ɪ/jɔ	-kʊɭa	-ʈʰɔ
-ɟɔnɪt̪ɔ	-gʊɖɪkɔ	- gʊɖɪkɔrɔ	-aɭʊ	-ɪa	-ejɔ	-kʰɒrɔ	-t̪ɔrɔ
-mʊkt̪ɔ	-re	-sɔmʊhɔ	-aɳi	-ɪla	-ɪsʈʰɔ	-kʰana	-t̪ɔmɔ
-gɔt̪ɔ	-rʊ	-bɔrgɔ	-ari	-ɪt̪a	-ʊarɔ	-cɪt̪	-t̪a
-a	-ʈʰarʊ	-d̪ɔɭɔ	-ɪ	-ɪma	-ʊria	-ʈa	-t̪ɔʔ
-d̪arɔ	-bɔnt̪ɔ	-bin	-ɭɔ	-ʊaɭɔ	-ʊɳɪ	-ɔnt̪ɪ	-aɳɪa
-pɔɳɔ	-bɔt̪i	-baɟ	-saɭi	-manɔ	-ʊɳa	-ɔnt̪ɔ	-ɔɳɔ
-bɔ	-bɔt̪	-rɔ	-sat̪	-rɔ	-ʊkɔ	-ɔɳɪ	-ɔɳa
-ɔɳijɔ	-ɪbɔ	-ɪt̪ʰɪba	-bad̪i	-hinɔ	-mɔjɔ	-bʰʊkt̪ɔ	-nɪrɔt̪ɔ
-ɪla	-ɪba	-ʊt̪ʰɪba	-siɭɔ	-d̪ajɔkɔ	-mɔji	-ɟɒgjɔ	-rɔhɪt̪ɔ

Table. 9. Untokenized Suffixes in Odia

3.4. Proposed Solutions for Annotation Scheme

This section provides necessary solutions that can be conducted to overhaul the BIS tagset for ILs incorporating some labels for other orphan categories.

3.4.1. The Case of Adverbs

There are two significant issues concerning the case of adverbs: first, the issue pertaining to the BIS tagset and second, related to the description of them. So far as the first issue is concerned, the interrogative adverb needs to be incorporated in the BIS tagset.

For instance,

If one asks you a question, “ How did you do the work?”

One will definitely answer it as “I did it nicely.”

Similarly, if one asks you the same question in Hindi, then you will reply it in the following way.

mɛne\PR_PRP ɪskɒ\DM_DMD ɔccʰe\RB se\RB kɪja\V_VM_VF |\RD_PUNC

Analoguously, one will answer it this way as mentioned below in Odia.

mɔ̃\PR_PRP ehɑkʊ\DM_DMD bʰɔlɔ\RB bʰɑbɔre\RB kɔlɪ\V_VM_VF |\RD_PUNC

In the foregoing, it can be concluded that, the answer of the question of how in any language will be a manner adverb always. This adverb can neither be ascribed to the interrogative pronoun nor can it be even related to the interrogative demonstrative prescribed by the BIS tagset; the only two interrogative tags in the scheme. Therefore, as a suggestion a tag of WRB[58] from the Penn Treebank or QRB could be incorporated within the tagset so as to include this unnoticed phenomenon of interrogative adverbs.

The second issue deals with the linguistic distribution of adverbs in any Indian language. It has been mentioned that one needs to follow the lexical approach while annotating the parts of speech under the BIS scheme. Thus, there is discrepancy with respect to the decision of labeling the manner adverbs as RB and going in favour of the morphological approach. Because, if one follows the morphological approach, many of the manner adverbs will have different improbable tags which may not treat them as manner adverbs.

Examples bʰɔlɔ\JJ bʰɑbɔre\N_NN ‘properly’

sɔpʰɔɭɔt̪ɑ\N_NN purbɔkɔ\JJ ‘successfully’

sɔhɔɟɔ\JJ rupɔre\NN_NN ‘easily’

Let us suppose that one will take recourse to the morphological approach while dealing with such expressions to avoid inconvenience of disambiguation problems and affected accuracy rate of the statistical tagger. But in the ILCI document, it has further been mentioned that one needs to annotate the adverbs of single word as manner adverbs: /t̪eʐ/, /ɟəld̪ɪ/, /d̪ʰɪre/ etc. (ILCI, 2010). Still one will face with the issue of ambiguity, if one annotates these as adverbs. The reason of the fact that same words can be used both as adjectives and adverbs. So, the issue of ambiguity is not a bigger deal than compromising on the linguistic information for annotating manner adverbs.

[58] WRB is also the label used by the ILMT tagset

Because, somehow or the other, the issue of ambiguity will remain intact in the field of machine learning and to achieve a cent percent accuracy, disambiguating all the ambiguity, is a mammoth task and next to impossible.

3.4.2. The Case of Demonstratives and Pronouns

Demonstratives also have possessive forms and there are a large number of such sort of data in the Odia corpus collected under the ILCI Project. Similarly, one can incorporate possessive pronouns into the scheme as the number of possessive forms in the corpora is quite large. Demonstratives having the possessive forms are mentioned below:

Examples egʊɖɪkɔrɔ ‘of these’

segʊɖɪkɔrɔ ‘of those’

ehɑrɔ ‘of this or it’

t̪ɑhɑrɔ ‘of that’

et̪ʰɪrɔ ‘of this’

set̪ʰɪrɔ ‘of that’

et̪ʰɪrɔ ‘of this’

set̪ʰɪrɔ ‘of that’

Therefore, it is suggested that under the BIS scheme, one can further make two more tags so as to ensure that these possessive forms get special treatments. If they are incorporated, then they may be having the tags like DM_POS for demonstrative possessive and PR_POS for pronoun possessive.

3.4.3. Compound Proper Nouns with Hyphenation

The compound proper nouns with hyphenation is one of the most interesting phenomena to deal with. These kinds of nouns originate from especially the transliterated data into Odia. In the following examples, it will become more obvious as the examples are the transliterated data in Odia from other languages.

Examples ɟɔn-ɪn-d̪ɪ-ʋɪlɖɔrnes\N_NNPC

χɪt̪ɑb-e-hɪnd̪\N_NNPC

The data in the form of compound expression demonstrated above can not be presented by separating all the parts of the proper noun. Besides, it does not fall into any of the category prescribed in the BIS. So, the tag of compound proper noun N_NNPC can be borrowed from the ILMT tagset.

3.4.4. Punctuations

As discussed earlier in the preceding section, it is clear that punctuations perform several functions other than their canonical functions. For instance, the hyphen functions as co-ordinator (co-ordinating words, phrases and clauses), list item markers, section headers, for frozen expressions etc. Hence, they need special attention when one is using them for POS annotation.

3.4.5. Reduplication

As stated already, it is indispensable that reduplication is one of the most important linguistic features of almost all the Indian languages and probably of many of the languages in the world. Since, BIS does not have a label of reduplication, it can be incorporated into the scheme from the ILMT tagset (Nainwani et al, 2012).

CHAPTER 4

4. COMPUTATIONAL FRAMEWORK, SYSTEM ARCHITECTURE & EVALUATION OF THE POS TAGGERS

This is one of the prominent chapters which is the soul of the current research. It has been categorized into six important sections. The first section mentions about the processes of developing linguistic resources for the training of the computational models. Thereafter, a detailed description of the SVM and CRF++ algorithms have been provided. The third section deals with the experimental set-ups which includes the feature extraction, configuration, training, testing, and evaluation. The fourth section contains the architecture for the online user interface of the tool. The following section provides a precise account of the technologies used for making the tool. Finally, the evaluation section contains four major sub-sections: the evaluation of the tagsets, the statistical models, error analysis of the models and proposed solutions.

4.1.Process of Developing Linguistic Resources

The linguistic resources have been developed by three major processes for modeling of the systems. They are annotation, validation, and tokenization. However, some of the other minor processes are not vividly dealt here.

4.1.1. Annotated Corpora

A total annotated corpus of approximately 2, 36, 793 numbers of tokens has been taken for the experiment during the training period. In both the seen and unseen data from the phase I, health and tourism comprise around 77k tokens. On the other hand, in phase II, entertainment, agriculture and literature comprise a total of 159k tokens data. It is obvious that during annotation of the data, the annotators commit errors and gradually, they increase their efficiency by getting into contact with a large number of other unique constructions. It is indispensable that the annotated data has to be processed at least once to maintain both the linguistic quality and for the cause of efficiency of the statistical taggers. The reason for error-prone nature of the data during the annotation phase is that there are many human annotators involved in the process and they cannot agree in several of the instances of judgment as to how should be a particular word tagged in a given context. Therefore, the next step which is followed is the validation process.

4.1.2. Validation Process

In this process, all the annotated data have been validated and thoroughly checked to ensure that there are no further errors or any undesired elements. The judgment to be taken at this stage is based on the inter-judgment between annotators. If there is a large-scale difference between the annotators, it has been decided based on the context of the word. In other words, when there is disagreement, the context has been given the utmost priority for judging. In the first example, there is only a control character placed just by the side of the main word which is an unwanted element and needs to be removed from the originally tagged file. Similarly, in the second and third examples, both the conventions are standard so far as the graphological convention is concerned and hence, are allowed. Lastly, if the tag of a word is wrongly labeled by the annotator, that has to be corrected in the file.

For example

- ଶୃଗାଳ ି= ଶୃଗାଳି
- ପଂଜରା or ପଞ୍ଜରା
- ଛଞ୍ଚାଣ\V_VM = ଛଞ୍ଚାଣ\N_NN

4.1.3. Tokenization

At this stage, the annotator checks the data and ensures that the data to be used for the experiment is really qualitative and each item is separated with a whitespace/tab space for the SVM and a tab space for the CRF++. The data has been tokenized using the Java Class Tokenizer. The automatic tokenizer tokenizes the data wrongly if there are unnecessary spaces or no any space between the two tokens. Therefore, one needs to be quite cautious dealing with the punctuations in the data.

For example

Input token ରାମରଘର,

Output token ରାମର ଘର , (separated by white spaces)

4.2.The Models of Odia Parts of Speech Tagger

This section has been categorized into two heads: the SVM and the CRF++ models.

4.2.1. Support Vector Machines

As stated by Marquez and Gimenez (2006), the SVM Tool is a simple and effective classifier and generator of sequential data based on Support Vector Machines. It is really suitable for 'practical NLP applications', robust, and flexible (for feature modeling) as it can process the data for automated tagging much faster in comparison to the other existing statistical taggers and demands almost less or no feature parameters to tune. By means of a rigorous experimental evaluation, one can give a concluding statement that SVM-based tagger is really beneficial for the annotation work of the parts of speech for any language. The only thing needed is a huge volume of the data. So far as the accuracy is concerned, as put forward by Marquez and Gimenez (2006):

> "The SVM-based tagger significantly outperforms the TnT tagger exactly under the same conditions, and achieves a very competitive accuracy of 97.2% for English on the Wall Street Journal corpus, which is comparable to the best taggers reported up to date. This version is implemented in Perl. A most efficient C++ version is currently available. The SVM light software implementation of Vapnik's Support Vector Machine (Vapnik, 1995) by Thorsten Joachims has been used to train the models".

In machine learning, support vector machines by Vapnik, as cited in Joachims (1999), are supervised learning models with associated learning algorithms that analyze data and recognize patterns. They are applied for classification and regression analysis. If a set of training examples is provided, by marking each of them as belonging to some of the categories, an SVM training algorithm prepares a model that labels tags to new input examples, which makes it a 'non-probabilistic binary linear classifier' (Marquez and Gimenez, 2006).

Given a set of N training examples $\{(x_1, y_1),\ldots, (x_N, y_N)\}$ where every instance x_i stands for a vector R^N and class label is $y_i \in \{-1,+1\}$. An SVM learns a linear hyperplane that separates the set of positive examples from the set of negative examples with maximal margin; the margin is defined as the distance of the hyperplane to the nearest of the positive and negative examples (Marquez and Gimenez, 2006) (see Fig. 5 below).

The linear separator is defined by two components: a weight vector w (with one component for each feature), and a bias b which stands for the distance of the hyperplane to the origin. The classification rule of an SVM is:

$$\text{sgn}\ (f\ (x, w, b)) \qquad (1)$$

$$f\ (x, w, b) = <w \cdot x> + b \qquad (2)$$

being x the example to be classified. In the linearly separable case, learning the maximal margin hyperplane (w, b) can be stated as a convex quadratic optimization problem with a unique solution: minimize ||w||, subject to the constraints (one for each training example):

$$y_i\ (<w \cdot x_i> + b) \geq 1 \qquad (3)$$

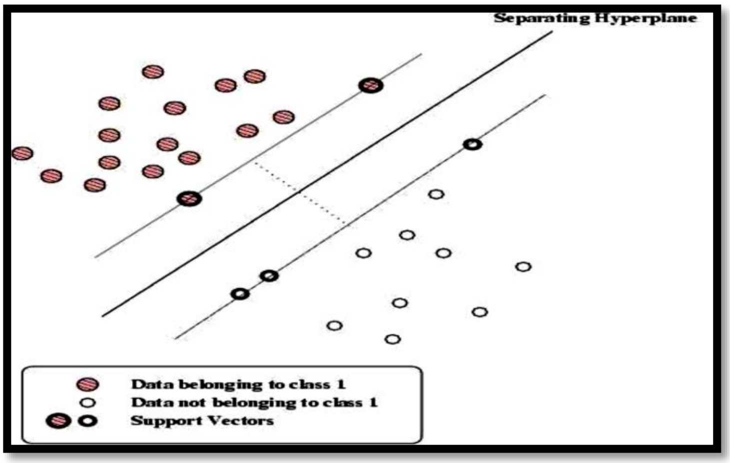


Fig. 5. SVM Classifier (classifying negative and positive examples)

Adapted from Gimenenez and Marquez (2006)

Given some training data $\mathcal{D}$, a set of n points of the form

$$\mathcal{D} = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in \mathbb{R}^p,\ y_i \in \{-1, 1\}\}_{i=1}^{n} \qquad (4)$$

So, they have developed the system using SVM (Joachims, 1999), which performs classification by constructing N-dimensional hyperplane that optimally separates data into two categories.

4.2.2. Conditional Random Fields

As discussed in (Lafferty et al., 2001), in what follows, X refers to any random variable among sequences of data to be annotated, and Y suggests a random variable out of corresponding label sequences of the data. All components namely Y_i of Y are assumed to range over a finite label alphabet Y. For example, X might range over natural language sentences and Y may range over POS taggings of those given sentences, with Y the set of possible POS annotation labels. “The random variables X and Y are jointly distributed”, but in a discriminative framework one creates a conditional model based on the conditional probability p(Y|X) from observation which is paired and annotate data sequences, and do not directtly model the marginal p(X).

As defined by (Lafferty et al., 2001),

Let G = (V, E) be a graph such that $Y = (Y_v)_{v \in V}$, so that, Y is indexed by the vertices of G. Then (X, Y) is a conditional random field in case, when conditioned on X, the random variables Y_v obey the Markov property with respect to the graph:

$p(Y_v|X,Y_w,$w is not equal to v$) = p(Y_v|X,Y_w,w \sim v)$, (where w ~ v means that w and v are neighbors in G)

Thus, a CRF++ is a globally conditioned random field on the observation X. They have assume that the graph G is fixed. In the simplest and most important example for modeling sequences, G is a simple chain or line: G = (V = {1, 2,...m}, E = {(i, i + 1)}).

X may also have a natural graph structure; yet in general it is not necessary to assume that X and Y have the same graphical structure, or even that X has any graphical structure at all. However, in they are concerned with sequences $X = (X_1, X_2,..., X_n)$ and $Y = (Y_1, Y_2,...,Y_n)$.

If the graph G = (V, E) of Y is a tree (of which a chain is the simplest example), its cliques are the edges and vertices. Therefore, by the basic formulation of random fields (Hammersley & Clifford, 1971), the joint distribution over the label sequence Y given X has the form

$$p\theta(y|x) \propto \qquad (5)$$

$$\exp\left(\sum_{e\in E,k} \lambda_k\, f_k(e, \mathbf{y}|_e, \mathbf{x}) + \sum_{v\in V,k} \mu_k\, g_k(v, \mathbf{y}|_v, \mathbf{x})\right)$$

where x is a data sequence, y a label sequence, and y|S is the set of components of y associated with the vertices in subgraph S.

One assumes that the features f_k and g_k are given and fixed. For example, a Boolean vertex feature gk might be true if the word X_i is upper case and the tag Y_i is "proper noun."

The parameter estimation problem is to determine the parameters θ = (λ_1, λ_2,...; μ_1, μ_2,...) from training data

$$\mathcal{D} = \{(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})\}_{i=1}^{N} \qquad (6)$$

with empirical distribution p̃(x,y). They describe an iterative scaling algorithm that maximizes the log-likelihood objective function$\mathcal{O}(\theta)$:

$$\begin{aligned} \mathcal{O}(\theta) &= \sum_{i=1}^{N} \log p_\theta(\mathbf{y}^{(i)} \mid \mathbf{x}^{(i)}) \\ &\propto \sum_{\mathbf{x},\mathbf{y}} \widetilde{p}(\mathbf{x}, \mathbf{y}) \, \log p_\theta(\mathbf{y} \mid \mathbf{x}) \; . \end{aligned}$$

As a particular case, one can construct an HMM-like CRF++ by defining one feature for each state pair (y',y), and one feature for each state-observation pair (y,x):

$$\begin{aligned} f_{y',y}\,(<u,v>, \mathbf{y}|_{<u,v>}, \mathbf{x}) &= \delta(\mathbf{y}_u, y')\,\delta(\mathbf{y}_v, y) \\ g_{y,x}\,(v, \mathbf{y}|_v, \mathbf{x}) &= \delta(\mathbf{y}_v, y)\,\delta(\mathbf{x}_v, x) \end{aligned} \qquad (7)$$

The corresponding parameters $\lambda_{y',y}$ and $\mu_{y,x}$ play a similar role to the (logarithms of the) usual HMM parameters p(y'\y) and p(x\y). Boltzmann chain models (Saul & Jordan, 1996; MacKay, 1996) have a similar form but use a single normalization constant to yield a joint distribution, whereas CRFs++ use the observation-dependent normalization Z(x) for conditional distributions.

4.3.Experimental Set-ups for SVM and CRF++

This is one of the prominent sections computationally as it encapsulates feature extraction, configuration files, training, testing and evaluation data and its format for both the models.

4.3.1. Feature Extraction

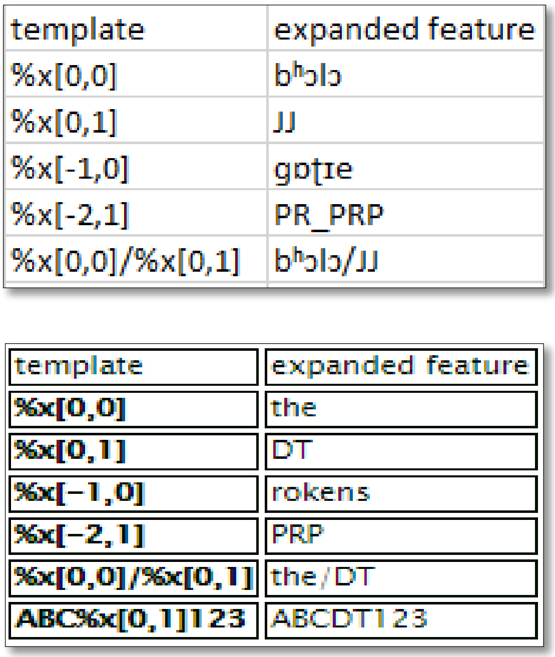
For both the models, simple features have been selected and several other features have been set to default mode[59]. The rationale for selecting simple features is

[59] please refer to the next section

that the salient linguistic features of the Odia words encoded in the affixes, agglutinating forms of the words, compound words, and morphophonemics have already been dealt with from linguistic perspective taking into consideration the guideline for annotation at the annotation phase. Since simple features have already been selected for SVM, so are they for CRF++.

4.3.2. Configuration Files for CRF++ and SVM

Unigram feature templates have been configured for the CRF++ model during training and testing. Each line in the below template file refers to one template. In each of the template, a special macro viz. %x[0,0] = %[row, column] is employed to represent an input data token. “In the template, ‘row’ specifies the relative position from the current focusing token and ‘col’ specifies the absolute position of the column”.[60]

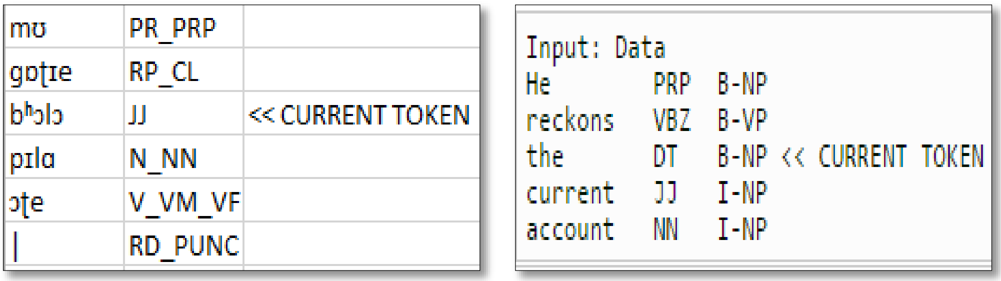


template	expanded feature
%x[0,0]	bʰɔlɔ
%x[0,1]	JJ
%x[-1,0]	gɒʈɪe
%x[-2,1]	PR_PRP
%x[0,0]/%x[0,1]	bʰɔlɔ/JJ

template	expanded feature
%x[0,0]	the
%x[0,1]	DT
%x[−1,0]	rokens
%x[−2,1]	PRP
%x[0,0]/%x[0,1]	the/DT
ABC%x[0,1]123	ABCDT123

Fig. 6. The Unigram Feature Templates for Odia CRF++ and the Adapted Template for English from ConLL.

[60] http://taku910.github.io/CRF++pp/

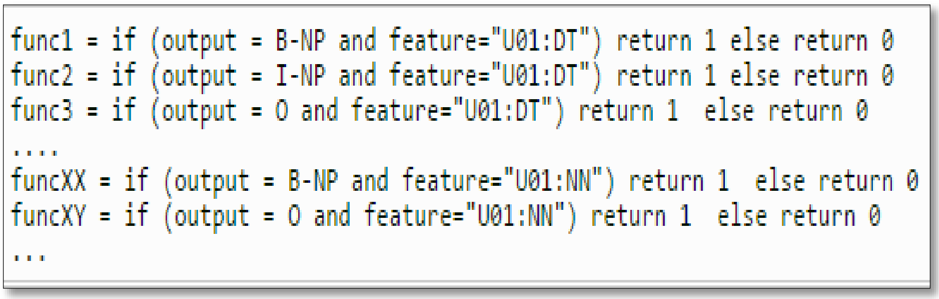


mʊ	PR_PRP	
gɒʈɪe	RP_CL	
bʰɔlɔ	JJ	<< CURRENT TOKEN
pɪlɑ	N_NN	
ɔʈe	V_VM_VF	
|	RD_PUNC	

```
Input: Data
He       PRP  B-NP
reckons  VBZ  B-VP
the      DT   B-NP << CURRENT TOKEN
current  JJ   I-NP
account  NN   I-NP
```

Fig. 7. The Description of the Feature Templates (Odia and English)

“This is a template to describe unigram features. When one gives a template "U01:%x[0,1]", CRF++ automatically generates a set of feature functions (func1 func2... funcN) like:”[61]

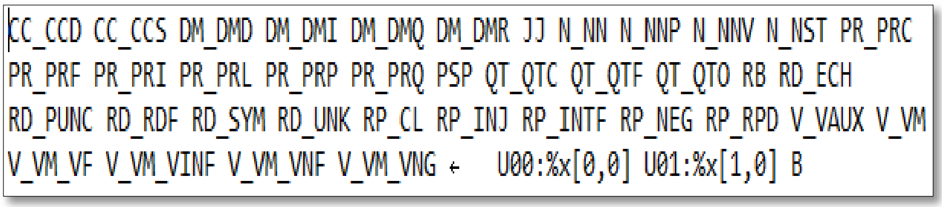


```
func1 = if (output = B-NP and feature="U01:DT") return 1 else return 0
func2 = if (output = I-NP and feature="U01:DT") return 1 else return 0
func3 = if (output = O and feature="U01:DT") return 1  else return 0
....
funcXX = if (output = B-NP and feature="U01:NN") return 1  else return 0
funcXY = if (output = O and feature="U01:NN") return 1  else return 0
...
```

Fig. 8. The Description of Unigram Functions

“The number of feature functions generated by a template amounts to (L * N), where L is the number of output classes and N is the number of unique string expanded from the given template”.[62]

- **Configuration File for CRF++:**



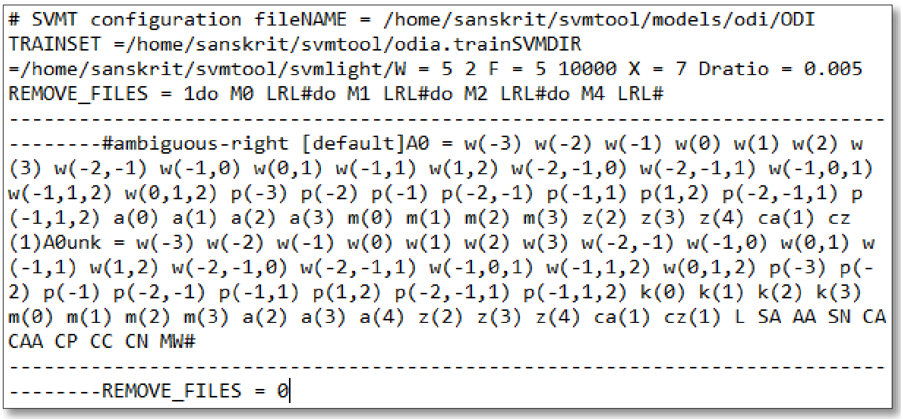
```
CC_CCD CC_CCS DM_DMD DM_DMI DM_DMQ DM_DMR JJ N_NN N_NNP N_NNV N_NST PR_PRC
PR_PRF PR_PRI PR_PRL PR_PRP PR_PRQ PSP QT_QTC QT_QTF QT_QTO RB RD_ECH
RD_PUNC RD_RDF RD_SYM RD_UNK RP_CL RP_INJ RP_INTF RP_NEG RP_RPD V_VAUX V_VM
V_VM_VF V_VM_VINF V_VM_VNF V_VM_VNG ↵  U00:%x[0,0] U01:%x[1,0] B
```

Fig. 9. The Configuration File for the CRF++

[61] http://taku910.github.io/CRF++pp/
[62] Ibid.

- **Configuration File for SVM:**

The configuration file that has been used during SVM learning phase contains medium verbose (-V 2) and the mode of learning and tagging has been set to left-right-left (LRL). And the rest of the features like sliding window, feature set, feature filtering, model compression, C parameter tuning, dictionary repairing and so on have been set to the default mode. The following feature template has been configured for the known and unknown ambiguous words.



```
# SVMT configuration fileNAME = /home/sanskrit/svmtool/models/odi/ODI
TRAINSET =/home/sanskrit/svmtool/odia.trainSVMDIR
=/home/sanskrit/svmtool/svmlight/W = 5 2 F = 5 10000 X = 7 Dratio = 0.005
REMOVE_FILES = 1do M0 LRL#do M1 LRL#do M2 LRL#do M4 LRL#
---------------------------------------------------------------------------
--------#ambiguous-right [default]A0 = w(-3) w(-2) w(-1) w(0) w(1) w(2) w
(3) w(-2,-1) w(-1,0) w(0,1) w(-1,1) w(1,2) w(-2,-1,0) w(-2,-1,1) w(-1,0,1)
w(-1,1,2) w(0,1,2) p(-3) p(-2) p(-1) p(-2,-1) p(-1,1) p(1,2) p(-2,-1,1) p
(-1,1,2) a(0) a(1) a(2) a(3) m(0) m(1) m(2) m(3) z(2) z(3) z(4) ca(1) cz
(1)A0unk = w(-3) w(-2) w(-1) w(0) w(1) w(2) w(3) w(-2,-1) w(-1,0) w(0,1) w
(-1,1) w(1,2) w(-2,-1,0) w(-2,-1,1) w(-1,0,1) w(-1,1,2) w(0,1,2) p(-3) p(-
2) p(-1) p(-2,-1) p(-1,1) p(1,2) p(-2,-1,1) p(-1,1,2) k(0) k(1) k(2) k(3)
m(0) m(1) m(2) m(3) a(2) a(3) a(4) z(2) z(3) z(4) ca(1) cz(1) L SA AA SN CA
CAA CP CC CN MW#
---------------------------------------------------------------------------
--------REMOVE_FILES = 0
```

Fig. 10. The Configuration File for SVM

4.3.3. Training Data Sets for CRF++ and SVM Models

The same number of data has been used for training both the CRF++ and SVM models. In the table, the first column demonstrates the phases of the ILCI Project from where the data has been taken in different domains (barring literature domain). The second column explains about the domains of the data. The third and fourth columns represent the seen and unseen training data respectively for the experiment of the taggers.

The tabulated data demonstrates that for both the models, 1, 41, 709 and 2, 36, 793 numbers of tokens have been taken for the experiment of seen and unseen data respectively. In both the seen and unseen data from the phase I, health and tourism comprise around 77k tokens. On the other hand, in phase II, entertainment, agriculture and literature comprise 64k tokens seen data while they cover 159k tokens unseen data. As represented in the table, it can be stated that the first phase data for both seen and unseen sets have been the same number each. Similarly, in the second phase the seen and unseen data for agriculture has been the same. So, the unseen data which has been increased during the training period belongs to the domains of entertainment and literature.

Training Data Sets	Domains	Tokens seen	Tokens unseen
First Phase	Health	46, 785	46, 785
	Tourism	30, 987	30, 987
Second Phase	Entertainment	13, 834	30, 929
	Agriculture	29, 470	29, 470
	Literature	20, 633	98, 622
Total Tokens		1, 41, 709	2, 36, 793

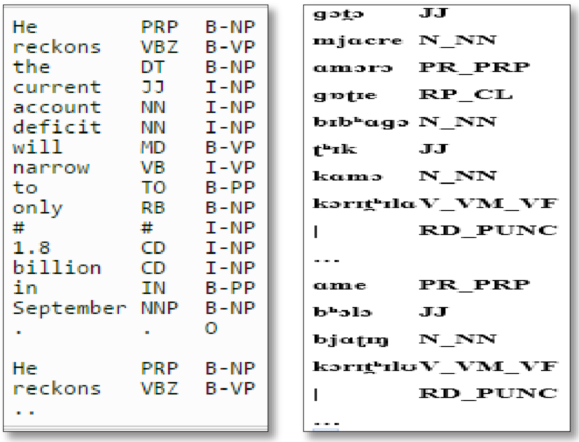
Table. 10. Training Data Sets for Odia Taggers

4.3.3.1.Training the CRF++ Model[63]

"CRF++ is a simple, customizable, and open source implementation of Conditional Random Fields for segmenting/labeling sequential data". For CRF++ to work properly both the training and test file need to be prepared in a particular uniform format. Broadly speaking, both of these files comprise of multiple tokens or words having multiple columns. Each token or word needs to be represented in one line, with the columns separated by white space or a tab character. A sequence of tokens or words becomes a sentence and to identify the boundary or break between sentences, an empty line is put. One can give as many columns as they like depending upon the work they are assigned to. However, the number of columns must be fixed through all tokens and throughout the file for consistency reason. If one is dealing with the parts of speech tagging and chunking, they can have only two columns and 3 columns respectively. Furthermore, there are some kinds of "semantics" governing among the columns of the tokens. For instance, 1st column stands for 'word', second column corresponds to 'POS tag', and third column is 'sub-category of POS' or 'chunk tag' and so on. There are 4 major parameters to control the training condition: -a, -c, -f, and –p.

[63] http://taku910.github.io/CRF++pp/

(Data for CoNLL shared task)



```
He          PRP  B-NP
reckons     VBZ  B-VP
the         DT   B-NP
current     JJ   I-NP
account     NN   I-NP
deficit     NN   I-NP
will        MD   B-VP
narrow      VB   I-VP
to          TO   B-PP
only        RB   B-NP
#           #    I-NP
1.8         CD   I-NP
billion     CD   I-NP
in          IN   B-PP
September   NNP  B-NP
.           .    O

He          PRP  B-NP
reckons     VBZ  B-VP
..
```

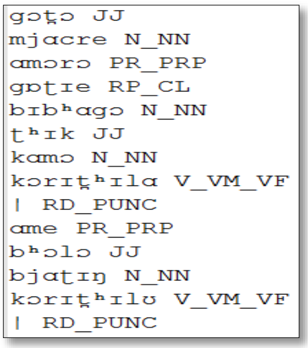
```
gɔt̪ɔ       JJ
mjacre     N_NN
amɔrɔ      PR_PRP
gɒt̪ɪe       RP_CL
bɪbʰagɔ    N_NN
t̪ʰɪk       JJ
kamɔ       N_NN
kɔrɪt̪ʰɪla V_VM_VF
|          RD_PUNC
...
ame        PR_PRP
bʰɔlɔ      JJ
bjat̪ɪŋ     N_NN
kɔrɪt̪ʰɪlʊ V_VM_VF
|          RD_PUNC
...
```

Fig. 11. Training Data Format for CRF++ Data[64]

4.3.3.2.Training the SVM Model

Given a training set of examples (either annotated or unannotated), it is responsible for the training of a set of SVM classifiers. To do so, it makes use of SVM–light7, an implementation of Vapnik’s SVMs in C, developed by Thorsten Joachims. The SVM light software implementation of Vapnik’s Support Vector Machine in 1995 by Thorsten Joachims has been used to train the models (Joachims, 1999). For training, 72k tokens (first phase) and 59k tokens (second phase) data have been used.

Training data needs to be in a columned manner, i.e. ‘a token per line corpus’, in a sentence-sentence fashion. The very first field needs to represent the token while the second one is the corresponding tag. The rest of the columns are not so necessary for parts of speech tagging and can contain additional information (Marquez and Gimenenez, 2006). For example,



```
gɔt̪ɔ JJ
mjacre N_NN
amɔrɔ PR_PRP
gɒʈɪe RP_CL
bɪbʰagɔ N_NN
ʈʰɪk JJ
kamɔ N_NN
kɔrɪt̪ʰɪla V_VM_VF
| RD_PUNC
ame PR_PRP
bʰɔlɔ JJ
bjaʈɪŋ N_NN
kɔrɪt̪ʰɪlʊ V_VM_VF
| RD_PUNC
```

[64] The training data format for English has been adapted from the ConLL shared task

Fig. 14. The Training Data Format for Odia SVM

SVMTlearn behaviour is easily adjusted through a configuration file. These are the currently available options:

- Sliding window (size and core position)
- Feature set (word features, POS features, orthographic features, 'multiple-column' features)
- Feature filtering (count cutoff and max mapping size)
- SVM model compression
- C parameter tuning
- Test [against a test set or via cross-validation]
- Dictionary repairing (either heuristically and/or based on a correction list)
- Ambiguous classes (may be optionally provided)
- Open classes (may be optionally provided)
- Backup lexicon (may be optionally provided)

4.3.4. Testing the CRF++ and SVM Models

The same testing data sets have been used for testing both the models. The tabulated data demonstrates that first phase seen data consists approximately of 31k tokens while the unseen set comprises 47k tokens.

Testing Data Sets	**Domains**	**Tokens seen**	**Tokens unseen**
First Phase Data	Health	15, 935	32, 691
	Tourism	15, 442	14, 407
Second Phase Data	Entertainment	13, 834	18, 463
	Agriculture	29, 470	17, 885
	Literature	20, 633	45, 200
Total Tokens		95, 314	1, 28, 646

Table. 11. Testing Data Sets for Odia Taggers

The second phase seen data comprises around 64k tokens while the unseen comprises 82k tokens. The unseen data in the domains of health, entertainment and literature has been increased in comparison to the seen data. The total number of token data in the seen and unseen comprises around 95k and 129k respectively.

4.3.4.1.Testing the CRF++ Model

The same procedure as used for training the CRF++ data is employed for the testing.

4.3.4.2.Testing the SVM Model

As put forward by Gimenenez and Marquez (2006), the SVMTagger annotates the parts of speech of a group of words, if provided the path to a previously learned SVM model file including the dictionary which is automatically generated at the training phase. For that matter, the input file must confirm to the SVM standard i.e. one token per line. The automated annotation process takes place on-line showing a sliding window which presents feature context to be selected at every decision by the tagger. The tagger performs the parts of speech tagging in a sentence by sentence fashion and a standard input/output system; with the token is expected to be the first column and the tag to be the second one.

These are some of the currently available options for the user.

- Tagging scheme (greedy/sentence-level)
- Tagging direction (left-to-right, right-to-left, or both)
- One pass / Two passes
- SVM Model Compression
- Get all predictions (not only the winner)
- Use of a softmax function to transform predictions into probabilities
- Backup lexicon (may be optionally provided)
- Lemma lexicon (may be optionally provided)
- Use of a Levenshtein Distance module to enrich the dictionary taking in account the input

4.3.5. Evaluating the CRF++ and SVM Models

This sub-section contains the evaluation of both the models.

- **The CRF++ Model:**

The last column is given (estimated) tag. If the 3rd column is true answer tag, one can evaluate the accuracy by simply seeing the difference between the 3rd and 4th columns. Taking into consideration the verbose level and the N-best outputs or through the precision and recall method, the CRF++ can be evaluated. For the present research, the latter method of evaluation has been adhered i.e. the precision and recall. For evaluation, the conleval.pl[65] Perl file has been used.

- **The SVM Model:**

As discussed by Gimenenez and Marquez (2006), the SVMTeval evaluates the performance of the statistical tagger in terms of accuracy, if provided the predicted tagging output of the tagger and its corresponding gold standard data. It is a quite useful feature for "the tuning of the system parameters, like the C parameter, the feature patterns and filtering, the model compression et cetera".

The evaluation tool provides different sets of words: known vs unknown, ambiguous vs unambiguous, ambiguous known words vs unambiguous known words. Further, it provides lists of words commonly 'sharing the same degree of ambiguity' with various levels of ambiguity classes. These features are created automatically based on the morphological dictionary that it generates during training time. It provides the evaluation results in the following lines.

- A brief report on overall accuracy
- A comparison of known vs unknown words and ambiguous and unambiguous words
- Grouping of words as per their level of disambiguation complexity
- Grouping of words as per their class of ambiguity
- Presentation of the accuracy from the perspective of parts of speech

4.4.The Online User Interface and The Architecture of the Odia Statistical Taggers

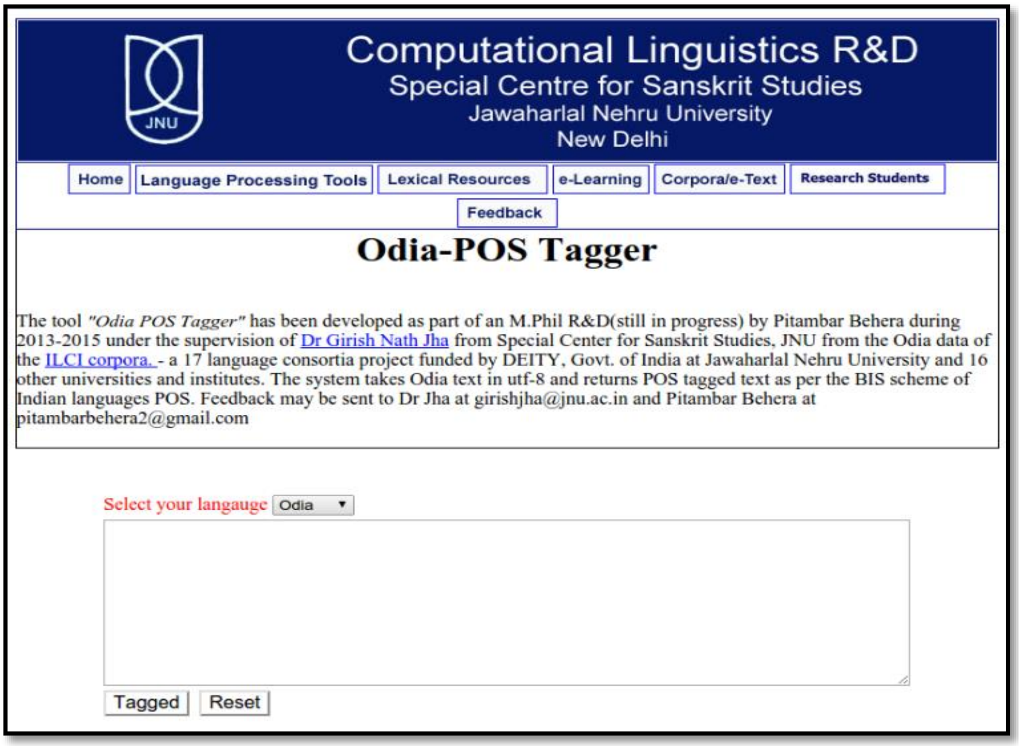
To create any program or NLP application tool, one needs resources such as manpower, programmer, software, and hardware. In this case of preparing Odia SVM

[65] http://lcg-www.uia.ac.be/conll2000/chunking/

Tagger, resources like web-technology system, software and skilled programmers are available to make this task accomplished.

4.4.1. The User Interface and Architecture for Odia Taggers

The tool is a web-based platform which is implemented with the JSP code and run with Apache Tomcat on the server. It will be available online on the web of the Computational Linguistics Research and Development, SCSS, JNU with the link (http://sanskrit.jnu.ac.in/pos/odia.jsp). The online SVM/CRF++ Taggers for Odia (see Flow Chart Fig. 13) has the following interactive user interface structure.



Odia-POS Tagger

The tool *"Odia POS Tagger"* has been developed as part of an M.Phil R&D(still in progress) by Pitambar Behera during 2013-2015 under the supervision of Dr Girish Nath Jha from Special Center for Sanskrit Studies, JNU from the Odia data of the ILCI corpora. - a 17 language consortia project funded by DEITY, Govt. of India at Jawaharlal Nehru University and 16 other universities and institutes. The system takes Odia text in utf-8 and returns POS tagged text as per the BIS scheme of Indian languages POS. Feedback may be sent to Dr Jha at girishjha@jnu.ac.in and Pitambar Behera at pitambarbehera2@gmail.com

Fig. 14. The Online User Interface of Odia POS Taggers

4.4.1.1.Input Text

Firstly, a user provides the input files or sentences in Odia script to the online platform. The platform identifies only the UTF encoding of the raw input text to be processed. If a user encodes the input file with editor software other than the Unicode font it will not be identifiable by the Tool. However, there is no specific limit in terms of the quantity of the data to be given. There is no lower limit of the data to be encoded; it can be a single token.

4.4.1.2.Pre-processor

The pre-processor filters the input text and checks whether any unwanted components are not present in the same. If it finds out so, it either discards them from the input text or leaves as they were earlier during the input. For example, if it finds non-specified characters like the unwanted punctuations within the token or half-finished letters or any other ‘control characters’ (Choudhary, 2006), it leaves them as they are by labeling with a tag.

- Input token:

ମୃ?ଦଙ୍ଗ

- Output token

ମୃ?ଦଙ୍ଗ N_NN

4.4.1.3.Tokenization

The next step that the tool approaches to is that it tokenizes the input data which is encoded in a sentence-by-sentence fashion. Further, it tokenizes the input data wherever it finds two tokens separated by a white space. Thereafter, it converts the file with sentences to token-by-token fashion. The tokenizer used in the tool is the Java Class Tokenizer.

4.4.1.4.The SVM Tool/CRF++ Toolkit

Thirdly, the Tool forwards to the SVM tool/CRF++ Toolkit which is run by the SVM algorithm/CRF algorithm respectively. It reaches to the model and input files and implements them. At this important stage, the SVM processes the input data in two phases: the LR mode and the RL mode. Thereafter, it annotates them based on its previous learning and provides the output identifying the probable tag for the given input token. If one selects the SVM tagger to process, the SVM tool will annotate the input file. If one selects the CRF button, the toolkit starts processing the data based on its earlier training.

4.4.1.5.The POS-tagged Output

The quality of the output decoded by the tagger is based on the efficiency in the training data. To make the tagger more efficient, one needs to focus much on the training period. The output generated by the tagger is in a token-by-token fashion in each line. It solely depends upon the input file as to what will be the probable best output of the input data. If a user provides a phrase, the tagger provides the tagged phrase by tokenizing it, provided there are no punctuations or any control character in between or attached with the token.

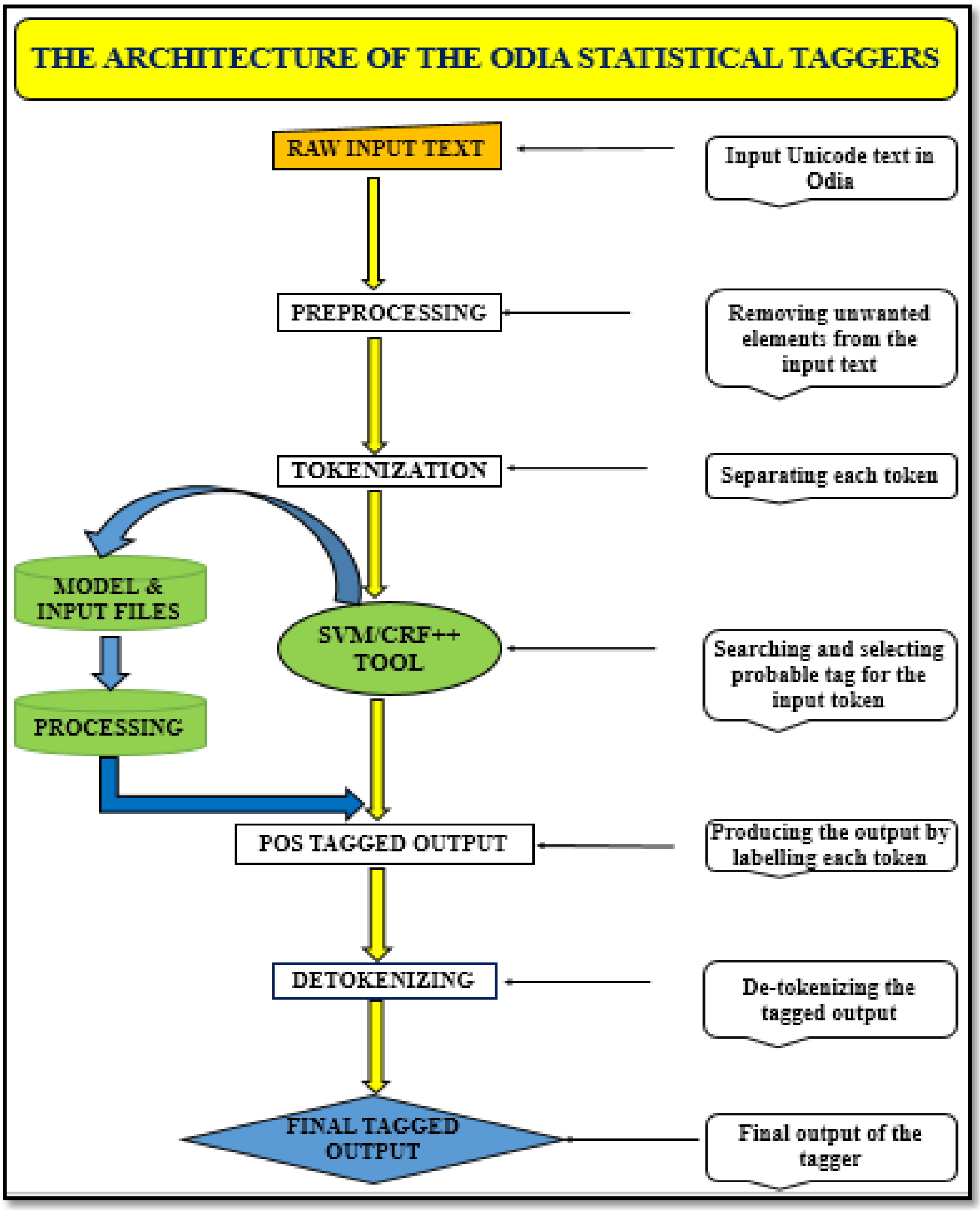

THE ARCHITECTURE OF THE ODIA STATISTICAL TAGGERS
RAW INPUT TEXT
Input Unicode text in Odia
PREPROCESSING
Removing unwanted elements from the input text
TOKENIZATION
Separating each token
MODEL & INPUT FILES
PROCESSING
SVM/CRF++ TOOL
Searching and selecting probable tag for the input token
POS TAGGED OUTPUT
Producing the output by labelling each token
DETOKENIZING
De-tokenizing the tagged output
FINAL TAGGED OUTPUT
Final output of the tagger

Fig. 13. The Architecture of the Odia POS Taggers

4.4.1.6.The De-tokenizer

The tokenizer tokenizes each linguistic element into token while the de-tokenizer detokenizes them into the previous order. So the tokenizer and the de-tokenizer are contrary to each other. Thus, the de-tokenizer converts the tagged output text into its tokenized forms; separating each token and tag with a white-space. Thereafter, the tool provides the final output.

4.5.Technology Used for Making the Tool

The front end data for the application has been developed applying Servlets, Java Server Page, and HTML. The JSP page has been UTF-8 enabled and supports the scripts of any language encoded with UTF-8. The online platform runs on the Apache Tomcat 4.0 which is a container Java Servlet and Java Server Pages and the back end data of the tool.

The data opens online in a web browser which is based locally on the user's computer. The URL opens the JSP file located on the host computer usually at the path given. The browser, with the help of the java-webserver, reads the odiasvm.jsp file. To understand the structure and functions of the said file, one needs to have a look at the following.

The technical environment of the application is as follows:

- Programming language used is Java
- Web-based Tools are Servlets and JSP
- Server used is Apache Tomcat 4.0

4.5.1. Apache Tomcat 4.0

Apache Tomcat 4.0 supports the web applications that are built for the Servlet 2.2 and JSP 1.1 specifications (applied with no change) which was officially announced on September 17.[66] This is developed in an open and participatory environment and released under Apache Software License. Tomcat is a web-based server software for developing and running Java server pages on a local host (Chowdhary, 2006).

[66] http://www.oracle.com/technetwork/java/javaee/servlet/index.html

4.5.2. JSP

Java server pages are html pages which use Java objects embedded in the html code. JSP technology is an extension of the servlet technology created to support authoring of HTML and XML pages. It makes the process easier to combine fixed or static template data with dynamic content. Even if one is comfortable in writing servlets, there are several compelling reasons to investigate JSP technology as a complement to their existing work. Java Server Pages are utilized in creating webpage content with the application of Java-written XML and scriplets[67].

4.5.3. Java Servlet Technology

A servlet is an application program, written in Java and executed on a java compatible web server. It is applied for enhancing and extending the Web servers. One of the reasons for it being user-friendly is that it is 'server and platform-independent'[68]. It can avail all the benefits of Java language like portability, performance, reusability, and protection.

> "A reference to a servlet appears in the mark-up for a web page, in the same way that a reference to a graphics file appears. The web-server executes the servlet and sends the results of the execution (if there are any) to the web browser as HTML text" (Chowdhary, 2006).

4.6.Evaluation

The evaluation section contains two important sub-sections: evaluation of the tagsets for ILs and the Odia taggers developed during this research.

4.6.1. Evaluation of the Tagsets

This sub-section discusses the need for a tagset, its types, and issues in designing tagset along with comparison of the tagset made for ILs. Annotated corpus of a language facilitates the research and development activities in the field of Natural Language Processing. Many of such corpora have been developed all around the world in general and especially in English. For annotation, one needs a tagset to follow which accounts both for linguistic appropriateness and consistent annotation. Considering the NLP

[67] http://www.serverwatch.com/news/article.php/1125001/Apache-Tomcat-40-Final-Released.html
[68] http://www.oracle.com/technetwork/java/javaee/servlet/index.html

scenario in ILs, it is not as advanced as English is. In India, the situation is not so conducive considering the existence of four different language families, viz., Austro-Asiatic, Dravidian, Indo-Aryan and Tibeto-Burman, out of which Dravidian and Indo-Aryan (IA) comprise the largest group of languages spoken in the sub-continent (Baskaran et al., 2008). Therefore, to make a common parts of speech annotation standard for all ILs dealing with each and every special nuance of them is a challenging task.

4.6.1.1.Issues in Framing Tagsets

As has been discussed by Bharti et al., (2006), there are many significant issues in designing a standard tagset for any parts of speech annotation. The issues become pertinent when the question comes to the preparation of annotation standard for Indian languages. Since India is the homeland for more than four diverse language families, the designing of tagset has proved to be a mammoth task. There are linguistic issues such as finiteness vs coarseness, morphological vs syntactic, and new tags vs existing tags from a tagger.

4.6.1.1.1. Finiteness vs Coarseness

This issue originates from annotation process as to which approach one has to adhere to: ‘fine grained’ linguistic knowledge or ‘coarseness’. In other words, whether one has to account for finer parts of speech features or not.

In the below-mentioned example,

mɑnkɔɖɔmɑne gɔcʰɔgʊɖɪkɔre bɔsɪcʰɔnt̪ɪ “the monkeys are sitting on the trees”, both the nouns used as arguments are agglutinating.

- gɔcʰɔgʊɖɪkɔre ‘at/in/on the trees’ (common noun)
- gɔcʰɔ-gʊɖɪkɔ-re ‘at/in/on the trees’ {common noun + plural marker + locative case marker}

In the above-mentioned examples, the first follows only one level of fineness and the rest of the morpho-syntactic information are muted whereas the second is marked for the type of noun, plurality, and locative case marker. In other words, the first is coarse and the second is the fine-grained. Although one can argue in favour of having less number of tags which facilitates the machine learning, but the point is should one compromise missing out the linguistic knowledge inherently ingrained in those exemplary sentences especially ‘in agglutinating languages like Tamil, Telugu and some other Ils’ (Bharti et al., 2006) including Odia.

4.6.1.1.2. Morphological vs Syntactic

At the lexical level, as the category of a given word may function differently than it functions at the level of phrase. In most of the ILs, the manner adverbs in an adeverbal phrase comprises of words from different other categories; noun an postposition being prominent among them .

For instance

- mʊ̃ bʰɔlɔ\JJ bʰɑbɔre\N_NN kɑmɔ kɔrɪbɪ “I will do the work properly.”
- mʊ̃ bʰɔlɔ\RB bʰɑbɔre\RB kɑmɔ kɔrɪbɪ “I will do the work properly.”

In the instances stated above, it is quite obvious that the first example follows the morphological approach whereas the second one adheres to the syntactic approach. Since a word has syntactic relevance, it is plausible to annotate based on the syntactic information it contains. This may further take us into difficulties. So to make consistency and ease of machine learning, one needs to annotate morphologically without taking any recourse to syntax, semantics and pragmatics. These kinds of linguistic information have to be taken care of at higher stages of NLP like chunking, parsing, anaphora resolution and so on as rightly pointed out by Bharti et al., in 2006.

4.6.1.1.3. New Tags vs Tags from a Standard Tagger

One of the important considerable points is to create an entirely new tagset or to modify some of the tags from a standard tagger and take it as the reference for the annotation job. The latter option seems to be better as the labels used by the established tagger may be familiar with the users and hence, can prove to be easier in incorporation. The tagset designed by the Penn Treebank is one of the most accepted and commonly used tagsets and many of the subsequent sets developed later have been variant forms of this e.g. Lancaster tagset. Similarly, the ILMT tagset has been modeled upon the Penn Treebank and also other tags have been added whenever found necessary (Bharti et al., 2006).

4.6.1.2.Types of Tagsets

There are basically two types of tagsets: one which takes into account the fine-grained information to be covered and the other which takes recourse to the word and its corresponding tag. The former is known as the hierarchical and the latter is called as the flat structure.

- **The Flat Structure:**

Flat tagsets are lists of 'mutually exclusive categories', easier to process as having no quite long list of independent labels, and difficult to modularize and scale across languages (Baskaran et al., 2008). In other words, there is one to one correspondence between the tag description and the label of the tag; this list is not extremely large. They are really difficult to be incorporated for other languages other than for what they are meant for because there is no any provision for feature reusability at the morphosyntactic stage. Furthermore, these tagsets are capable of handling granularity; although they are easier to be processed (Baskaran et al., 2008). "Most of the popular English tagsets (including UPENN, Brown, C5 and C7) and the existing IL tagsets (IIIT-H, AU-KBC) fall under this type (Baskaran et al., 2008)."

- **The Hierarchical Structure:**

On the contrary, hierarchical tagsets are structured in nature and they contain different layers of categories and sub-categories in a tree structure. They have less number of tags at the highest level in comparison to the flat type and at the lower levels, they have sub-categories. Generally, the morphosyntactic features of languages are captured at the lower levels. The hierarchical structure of it allows for inclusion or exclusion of labels according to demands of the language to be incorporated which is known as 'decomposability'. It helps make the tagset uniform and suitable for the incorporation of any other language into its framework (Baskaran et al., 2008).

4.6.1.3.Tagsets for ILs

Hardie (2004), as cited in (Baskaran et al., 2008), has stated that the early tagsets such as UPENN, Brown and C5 (tagsets for English) mainly emphasised on simple lists of tags corresponding to the morphosyntactic features, and varied hugely with respect to granularity in the nineteen seventies. At this point, CLAWS2 tagset (Santorini, 1987) had been developed which was based on the hierarchical structure. The publication of

EAGLES recommendations for morphosyntactic annotation of corpora (Leech and Wilson, 1996) was one of the earliest attempts to develop a common annotation guideline for several European languages.

Various tagsets have been developed in ILs: the ILMT tagset by IIIT Hyderabad, AU-KBC Tamil tagset, IL-POSTS by MSRI (Microsoft research India Pvt. Ltd., LDC-IL by CIIL, BIS tagset by DeitY for ILCI Project, JNU-Sanskrit tagset (JPOS) and Sanskrit consortium tagset (CPOS) (Chandra et al., 2014). Of these tagsets, AU-KBC Tamil tagset has been designed only to cater to the needs of Tamil language. So this tagset cannot be extended to incorporate other languages as such. Nonetheless, the other tagsets have been designed as uniform standards for all the ILs. The ILMT tagset is based on a flat structure while the other tagsets are hierarchical in nature.

4.6.1.3.1. The ILMT-IIIT Hyderabad Tagset

As cited in Bharti et al., (2006), the ILMT tagset has been developed by the IIIT Hyderabad and is modeled upon the Penn Treebank annotation scheme. It has 21 categories and 26 labels. In addition, it modifies some of the existing labels and introduces some new labels to accommodate the ILs wherever necessary.

The whole tagset can be divided into three groups: Group I, II and Group III, as divided by Chandra et al., (2014).

Group I contains tags that are similar to the Penn Treebank. For instance, the ILMT has directly incorporated the tags of Common noun (NN), proper noun (NNP), pronoun (PRP), adjective (JJ), adverb (RB), interjection (UH) etc.

Group II contains the tags that have been modified according to the suitability of the ILs. For instance, the Penn Treebank has the tag of W used before the tags of different question words. Similarly, the ILMT has the tag of the WQ which refers to all the question words.

Group III contains tags that are completely new and addresses the unique linguistic features of the ILs. The locative noun (N_NST), negative particle (NEG), common compound nouns (NNC) and proper compound nouns (NNPC) etc.

4.6.1.3.2. The IL-POSTS Tagset by MSRI

MSRI (Microsoft Research India Pvt Ltd) has developed the IL-POSTS tagset in 2008. It aims at providing a comprehensive tagset that captures as much morphosyntactic information as possible from parts of speech tagging of ILs. In total, there are 11 categories out of which 9 are branched and the rest two are non-branched. It has further 32 types and 18 attributes. The punctuations and residual categories are universal categories applicable for all ILs and thus, these are mandatory for any tagset derived from IL-POSTS framework. There are 18 attributes defined currently in the IL-POSTS tagset. These attributes are either binary or multi-valued in nature (Baskaran et al., 2008). "The guideline of this tagset contains about nine categories (Nouns, Pronouns, Verbs, Nominal Modifier, Demonstrative, Adverb, Particle, Punctuation, and Residual) which branches out in types (such as common, proper, verbal, and spatio Temporal) (Chandra et al., 2014)."

(Baskaran et al., 2008) has stated that the IL-POSTS framework contains a hierarchy of three levels:

- Categories are the top-level part-of-speech classes all of which are mandatory, that is, are generally universally applicable to all languages and thus, must be encapsulated in any tagset based on morphosyntactic information and derived from this framework.
- Types are considered to be important sub-classes commonly applicable to a majority of languages. Some types may be optional while some other could be mandatory for certain languages.
- Attributes are the in-depth linguistic and morphosyntactic features of Types and are optional, although in some cases they may be recommended.

The IL-POSTS framework recommends the use of decomposable tags.

For example "NC.sg.loc.n.n"

In the example instantiated, N stands for the category 'noun', C stands for the type 'common' and the attributes are specified 'sg.loc.n.n' which implies that 'sg' stands for the singularity, 'loc' stands for the locative case marker 'n.n' stands for the absence of the classifier and the emphatics (Baskaran et al., 2008).

Under the head of decomposability, there are broadly three governing principles as discussed by (Baskaran et al., 2008) for this tagset framework. They are as follows.

- Each of the Categories and Types is represented by a unique single letter or a two-letter combination of tags that are in uppercase.
- It has also been made sure that the resultant string after the concatenation of a Category and its Type is not exceeding the three characters mark.
- The Attribute values are also assigned 1 to 4 character letters or numbers that are of unique strings.

4.6.1.3.3. The BIS Tagset by DeitY for ILCI

The Bureau of Indian Standards (BIS) Tagset has recommended the use of a common tagset for the part of speech annotation of ILs. This tagset is a result of the POS Standardization Committee appointed by the DeitY, Govt. of India. Presently, this tagset is being used for parts of speech annotation by the ILCI Corpora Project under the TDIL Programme. It has a total number of 11 categorical labels at the top level and 39 fine-grained labels for the annotation. The tagset has been framed keeping in view both the fineness and coarseness and flat hierarchical structures in view.

4.6.1.3.4. LDC-IL Tagset

The tagset framework consists of 14 categories, 43 labels, and 16 attributes with five values. The categories are mandatory, the attributes are recommended and the values are optional. This tagset was framed in the year 2009 and used by the CIIL for the annotation work in ILs. It is a fine-grained tagset and contains a hierarchical structure.

Sl. No.	**Categories**	**ILMT-IIIT Hyderabad**	**IL-POSTS by MSRI**	**LDC-IL**	**BIS-ILCI**
1	**Noun**		N	N	N
		Common (NN)	Common (C)	Common (C)	Common (NN)
		Proper (NNP)	Proper (P)	Proper (P)	Proper (NNP)
		Locative (NST)	Spatio-temporal (ST)	Spatio-temporal (ST)	Spatio-temporal (NST)
		Nil	Verbal (V)	Verbal (V)	Verbal (NNV)
2	**Pronoun**	PRP	P	P	PR

			Pronominal (PR)	Pronominal (PR)	Personal (PRP)
			Reflexive (RF)	Reflexive (RF)	Reflexive (PRF)
			Relative (RL)	Relative (RL)	Relative (PRL)
			Reciprocal (RC)	Reciprocal (RC)	Reciprocal (PRC)
			Wh-word (WH)	Wh-word (WH)	Wh-word (PRQ)
			Nil	Nil	Indefinite (PRI)
3	**Demonstrative**	DEM	D	D	DM
			Absolute (AB)	Absolute (AB)	Deictic (DMD)
			Relative (RL)	Relative (RL)	Relative (DMR)
			Wh-word (WH)	Wh-word (WH)	Wh-word (DMQ)
			Nil	Nil	Indefinite (DMI)
4	**Verb**		V	V	V
		Main (VM)	Main (M)	Main (M)	Main (VM)
		Nil	Nil	Nil	Finite (VF)
					Non-finite (VNF)
					Infinitive (VINF)
					Gerund (VNG)
		Auxiliary (VAUX)	Auxiliary (A)	Auxiliary (A)	Auxiliary (VAUX)
5	**Adverb**	RB	A	A	Manner (RB)
		Nil	Manner (MN)	Manner (MN)	
			Location (LC)	Nil	Nil
6	**Adjective Or Nominal Modifier**	JJ	J	J	JJ
		Nil	Quantifiers (Q)	Quantifiers (Q)	Nil
			Nil	Intensifier (INT)	
7	**Participle**	Nil	L	L	Nil
			Adjectival (RL)	Relative (RL)	
			Adverbial (V)	Verbal (V)	
			Nominal (N)	Nil	
			Conditional (C)	Conditional (C)	
8	**Conjunction**	CC	C	Nil	CC
		Nil	Nil		Co-ordinator (CCD)
		Nil			Subordinator (CCS)

	Quotative	UT			Quotative (CCS_UT)
9	**Postposition**	PSP	PP	PP	PSP
10	**Particles**	Particles (RP)	C	C	RP
		Nil	Nil	Emphatic (EMP)	Default (RPD)
		nil	Classifier (CL)	Classifier (L)	Classifier (CL)
	Interjection	INJ	Interjection (IN)	Interjection (IN)	Interjection (INJ)
	Intensifier	INTF	Others (X)	Others (X)	Intensifier (INTF)
	Negation	NEG			Negation (NEG)
		Nil	Co-ordinator (CD)	Coordinating (CD)	Nil
			Subordinator (SB)	Subordinating (SB)	
			Nil	Delimitive (DLIM)	
				(Dis)Agreement (AGR)	
				Exclusive (EXCL)	
				Terminative (TERM)	
				Dubitative (DUB)	
				Similative (SIM)	
11	**Quantifiers or Numeral**	Q	Nil	NUM	QT
		General (QF)		Real (R)	General (QTF)
		Cardinal (QC)		Serial (S)	Cardinal (QTC)
		nil		Calendric (C)	nil
		Ordinal (QO)		Ordinal (O)	Ordinal (QTO)
		Classifier (CL)		Nil	nil
12	**Residuals**	Nil	RD	RD	RD
		Nil	Foreign word (F)	Foreign word (F)	Foreign (RDF)
	Symbol	SYM	Symbols (S)	Symbols (S)	Symbols (SYM)
	Echo-word	ECH	Others (X)		Echo-words (ECH)
	Unknown	UNK		UNK	Unknown (UNK)
		Nil			Punctuation (RD_PUNC)
	Reduplicative	RDP	Nil	RDP	Nil

13	**Punctuations**		PU	PU	Nil
15	**Compounds**	Compounds (*C)	Nil		
16	**Question Words**	Question Words (WQ)			

Table. 13. Tagsets for Indian Languages

4.6.1.4.Comparison of Tagsets

A tagset of a language should neither be too coarse nor should it be much fine-grained. If it becomes so coarse, the analysis is not much of use at the level of parts of speech. If it becomes so fine-grained, it will hamper the machine learning. Therefore, it becomes indispensable that the best tagset is the tagset which strikes a unique balance between the coarseness & fineness on one hand and flatness & hierarchy on the other.

In the above table, the first column contains the serial numbers, the second has the common parts of speech categories covered by all the tagsets, the third contains ILMT-IIIT Hyderabad tagset, the fourth contains IL-POSTS tagset by MSRI, the fifth has LDC-IL tagset, and the sixth contains BIS-ILCI tagset. In all the first rows of each category the top level category along with the tag has been provided which applies to all the tags for the sub-categories within the broader category. If a category or sub-category is not applicable to any other tagset, then it is marked ‘nil’.

In total, there are 11 categories out of which 9 are branched and the rest two are non-branched in MSRI. It has further 33 types and 18 attributes. The ILMT tagset has been developed by the IIIT Hyderabad and is modeled upon the Penn Treebank annotation scheme. It has 21 categories and 26 types. LDC-IL tagset framework consists of 14 categories, 43 labels, and 16 attributes with five values. BIS has a total number of 11 categorical labels at the top level and 39 fine-grained labels for the annotation. In all the tagsets, one thing which is common is that all of them more or less agree on the top-level categories, but there is a great disagreement in terms of treating the lower level types and sub-types. It seems that LDC-IL and the MSRI tagset frameworks are having many things in common: especially with regard to categories, types, and attributes.

ILMT tagset is based on the flat structure, although it tries to strike a balance between the coarseness and fineness. It has a word and tag corresponding annotation labels with no much sub-categorization in many categories barring some. Apart from the ILMT, other tagsets are based on the hierarchical structure and they are fine except the BIS. BIS is a tagset which is an amalgamation of both the structures: flat and the hierarchical. So, it is neither completely flat nor is it hierarchical totally. Some of the parts of speech are annotated at the phrase level in the ILMT framework whereas other tagsets do not use at all, for instance, the compound common and the proper nouns. The ILMT follows the Penn Treebank while the MSRI is modeled upon the EAGLES tagsets. LDC-IL and the BIS seem to have made improvements on both the existing tagsets in ILs.

Many tags at the top-level category among different sets are the same with some varying labels. Except the ILMT, the other tagsets share many things in common as all of them are based on the hierarchical schema. Unlike the other three tagsets, ILMT does not include the label for the verbal noun. Unlike others having the sub-types of pronouns and demonstratives, the ILMT contains only one category each. Unlike BIS which contains six types of verbs, others have only two types of verbs viz. the main and the auxiliary. Unlike the MSRI which contains two types of adverbs-manner and location, the others have only manner adverb. ILMT and BIS have the only category for adjective, but the other two have other modifiers like the quantifiers and intensifiers including adjective under the category of nominal modifiers. MSRI and LDC-IL have different types of participles under the category of the participles while the others do not have the category. A conjunction is a separate category in the BIS, ILMT, and the MSRI whereas the LDC-IL contains no any. The BIS has the types of the conjunctions and the other two have single labels. Further, in the MSRI the categories of coordinating and subordinating conjunctions are under the category of the particles. LDC-IL has vividly described the category of particles and categorizes it into twelve while the BIS and MSRI have five categories each; with the ILMT having one. Reduplication phenomenon is being captured by the ILMT and LDC-IL. Compounds and question words are the categories taken up by the ILMT only.

All the tagsets designed till date have really provided an important computational platform for ILs. The MSRI tagset has made the distinction quite clear between the infinitive /rəhnɑ/ ‘to stay’ verb and the gerundive /rəhne/ ‘staying’ forms of it. Thus, it can be averred that it is based on the morphosyntactic approach dealing with the parts of speech. Similarly, the BIS assigns parts of speech labels both morphologically and contextually.

4.6.2. Evaluation of the CRF++ and SVM Taggers

This sub-section contains a comparative evaluation of both the models.

4.6.2.1.Automated Evaluation

The automated evaluation has been conducted in terms of the overall accuracy, the unknown vs unknown words, ambiguous vs unambiguous words, and accuracy per POS category for both the models.

4.6.2.1.1. Overall Evaluation

Results are always compared to the most-frequent-tag (MFT) baseline (Giminenez and Marquez, 2006). The results are based on the automatic evaluation of the SVM Tagger for Odia by the SVMTeval tool. The following figure demonstrates that the first bar which stands for the known words has an accuracy rate of 99.27% while the bar of unknown words states that it is around 0.72%. The reason for low accuracy is that hits of the unknown words is less in number. The accuracy rate of the ambiguous words out of the total number of the evaluated data is 30.22%. The MFT baseline is 90.90%. Out of the total number of ambiguous tokens (30.22%), known ambiguous are 90.12% whereas the unambiguous known tokens account for 99.77%.

Snippet 1. Results of the SVM Tagger on the Seen Data

```
==========================================================
==============

EVALUATING </home/sanskrit/svmtool/odia.eval.output> vs.
</home/sanskrit/svmtool/odiagold.txt> on model
</home/sanskrit/svmtool/models/odi/ODI>...

..........10000..........20000..........30000..........40000..........50000..........60000..........7
0000..........80000...........88958 tokens [DONE]

* =================

TAGGING SUMMARY
======================================================

#TOKENS = 88958

AVERAGE_AMBIGUITY = 1.7482 tags per token
```

```
* -------------------------------------------------------------------------------------

#KNOWN = 99.2772% --> 88315 / 88958

#UNKNOWN = 0.7228% --> 643 / 88958

#AMBIGUOUS = 30.2289% --> 26891 / 88958

#MFT baseline = 94.9077% --> 84428 / 88958

* ================ KNOWN vs UNKNOWN TOKENS
============================================

              HITS TRIALS ACCURACY

              86157 88958 96.8513%

* -------------------------------------------------------------------------------------
```

Snippet 2. Results of the SVM Tagger on the Unseen Data

```
======================================================
=============

EVALUATING </home/sanskrit/svmtool/odia.eval.output> vs.
</home/sanskrit/svmtool/odiagold.txt> on model
</home/sanskrit/svmtool/models/odi/ODI>...
..........10000..........20000..........30000..........40000..........50000..........60000..........7
0000..........80000..........90000..........100000..........110000..........120000...120859
tokens [DONE]
* ================= TAGGING SUMMARY
====================================================
#TOKENS = 120859
AVERAGE_AMBIGUITY = 6.5408 tags per token
* -------------------------------------------------------------------------------------
#KNOWN = 86.4859% --> 104526 / 120859
#UNKNOWN = 13.5141% --> 16333 / 120859
#AMBIGUOUS = 26.5251% --> 32058 / 120859
#MFT baseline = 82.6393% --> 99877 / 120859
* ================= KNOWN vs UNKNOWN TOKENS
```

```
===============================================
HITS TRIALS ACCURACY
113113 120859 93.5909%
* -------------------------------------------------------------------------------------
```

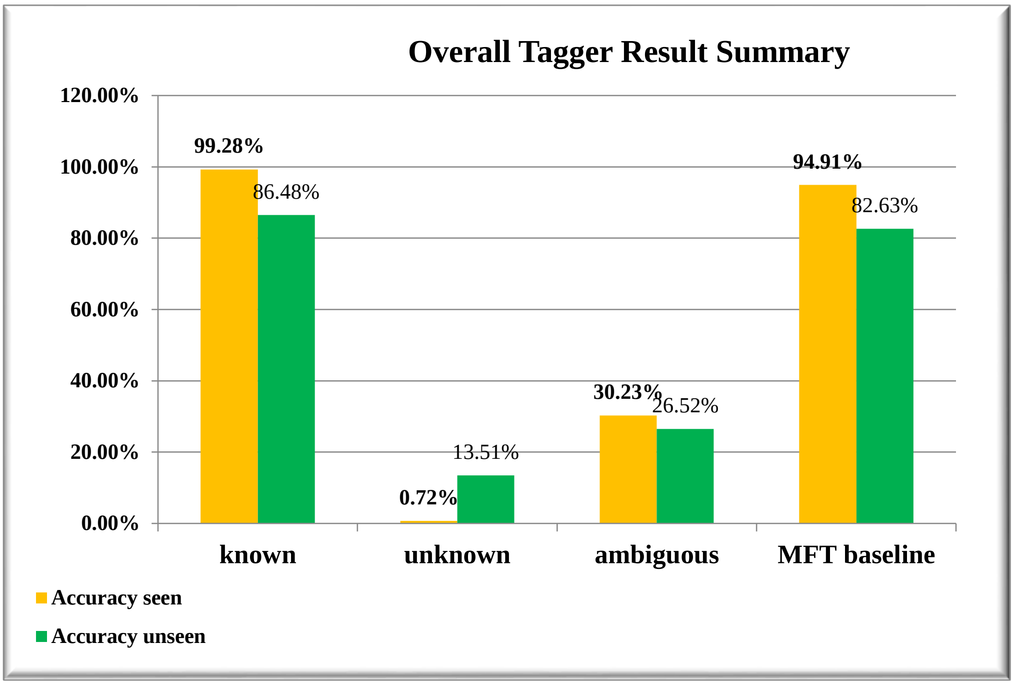


Chart. 1. Result Summary of the Odia SVM Tagger on the Seen and Unseen Data

4.6.2.1.2. Accuracy per Level of Ambiguity

The following tabulated data represents the class of ambiguity and along with their number of hits and trails, accuracy per level and MFT baseline during mechanical evaluation. There are nine ambiguity classes prepared by the machine. The highest ambiguous level that one of the levels has is the eighth one which includes some the ambiguity sets having the lowest accuracy rates. On the other hand, lowest ambiguous class is the first level.

Accuracy per Level of Ambiguity

Levels1/2	Hits1	Hits2	Trials 1	Trials 2	Accuracy 1	Accuracy 2	MFT 1	MFT 2
1	61287	71308	61424	72468	99.78%	98.39%	99.87%	98.39%
2	16036	19042	17605	20710	91.09%	91.94%	88.54%	89.85%

3	4969	6841	5571	7280	89.19%	93.96%	82.73%	87.51%
4	2660	3087	2960	3227	89.86%	95.61%	81.62%	89.33%
5	321	443	393	473	81.68%	93.65%	64.89%	84.35%
6	155	152	206	170	75.24%	89.41%	65.05%	75.29%
7	40	50	52	52	76.92%	96.19%	53.85%	80.76%
10/8	54	139	104	147	51.92%	94.55%	50.96%	93.87%
40/39	635	12051	643	16332	98.76%	73.78%	0.00%	0.00%

Table. 14. Accuracy per Level of Ambiguity (1= evaluation is done with the seen data, 2= evaluation is done with unseen data)

4.6.2.1.3. Accuracy per POS Category

This sub-section includes the accuracy per POS category for both the CRF++ and SVM models.

4.6.2.1.3.1.Accuracy per POS Category of the SVM Tagger

The following data represents the accuracy per level of parts of speech for the Odia SVM statistical tagger in both the seen and unseen domains.

As far as the seen data is concerned, of all the POS categories, the highest rate of accuracy is figured in the class of symbols with cent percent accuracy followed by the punctuation class, infinitive verb, finite verb, personal pronoun and so on. On the contrary, the lowest accuracy rate is registered in the class of the verbal noun as it does not get any hit, although it contains around hundred trials in the gold data. Besides verbal nouns, the lowest accuracy rate figured is in the class of wh-pronouns as it overlaps with the wh-demonstratives followed by the reciprocal and relative pronouns. In this way the overall accuracy rate of the tagger for Odia is 96.85 with a baseline MFT is 94.90, which is far better than any of the tagger reported so far in Indian languages.

On the other hand, in the domain of unseen data, highest accuracy is registered in the interrogative pronoun and foreign word with each having 100%. On the contrary, the lowest accuracy rate has been figured in the categories such as proper nouns, echowords, unknown and foreign words. Categories like negative, default particle, symbol, punctuation, postposition, coordination, and indefinite demonstrative are having the percentage of 99 mark and above.

Accuracy in Percentage per Part-of-Speech for the Odia SVM Tagger

POS	**Hit 1**	**Hit 2**	**Trial 1**	**Trial 2**	**Accuracy 1**	**Accuracy 2**	**MFT 1**	**MFT 2**
CC_CCD	2735	3925	2919	3944	93.6965	99.5183	92.2576	96.4757
CC_CCS	1025	1491	1156	1511	88.6678	98.6764	89.1869	98.0807
DM_DMD	2502	3394	2563	3447	97.6200	98.4624	97.5029	98.2594
DM_DMI	568	804	591	811	96.1083	99.1369	91.8782	97.4106
DM_DMQ	203	158	243	168	83.5391	94.0476	81.4815	91.6667
DM_DMR	409	454	428	476	95.5607	95.3782	93.2243	94.9580
JJ	6698	8922	7154	10404	93.6259	85.7555	94.4507	73.9908
N_NN	26400	38525	26763	40952	98.6436	94.0735	97.7581	75.4981
N_NNP	5854	5741	6313	8159	92.7293	70.3640	90.1948	50.7783
N_NNV	0	54	103	78	0.00	69.2308	81.5534	71.7949
N_NST	2237	2093	2270	2148	98.5463	97.4395	97.4890	92.7374
PR_PRC	9	12	20	13	45.0000	92.3077	55.00	92.3077
PR_PRF	365	436	373	440	97.8552	99.0909	97.8552	99.0909
PR_PRI	74	51	80	54	92.5000	94.4444	91.2500	88.8889
PR_PRL	34	89	55	95	61.8182	93.6842	69.0909	93.6842
PR_PRP	2241	1906	2251	1912	99.5558	99.6862	99.1559	98.3787
PR_PRQ	16	14	62	14	25.8065	100.0	37.0968	100.0
PSP	3552	3666	3666	3698	96.8903	99.1347	95.4173	97.1877
QT_QTC	1596	2971	1637	3005	97.4954	98.8686	97.2511	80.2995
QT_QTF	1455	1879	1529	1952	95.1602	96.2602	84.6959	89.4980
QT_QTO	304	563	305	580	99.6721	97.0690	99.0164	92.9310
RB	1102	1741	1298	1968	84.8998	88.4654	72.8814	79.4207

RD_ECH	9	2	17	3	52.9412	66.6666	47.0588	66.6667
RD_PUNC	10955	14466	10956	14510	99.9909	99.6968	99.9909	99.6968
RD_RDF	91	57	97	57	93.8144	100	93.8144	1.7544
RD_SYM	529	1087	529	1090	100	99.7248	99.6219	99.7248
RD_UNK	69	183	70	257	98.5714	71.2062	95.7143	10.5058
RP_CL	473	571	477	588	99.1614	97.1088	98.1132	89.7959
RP_INJ	72	16	74	17	97.2973	94.1176	94.5946	94.1176
RP_INTF	406	429	445	474	91.2360	90.5063	72.5843	72.7848
RP_NEG	698	888	702	890	99.4302	99.7753	50.00	99.7753
RP_RPD	1389	1329	1398	1334	99.3562	99.6252	78.6123	99.5502
V_VAUX	176	201	202	220	87.1287	91.3636	73.2673	79.0909
V_VM	786	924	850	1035	92.4706	89.2754	78.00	56.1353
V_VM_VF	8043	9528	8074	9615	99.6161	99.0952	99.0835	93.0629
V_VM_VINF	1866	2512	1873	2678	99.6263	93.8013	97.9178	90.8514
V_VM_VNF	857	1444	1017	1623	84.2675	88.9710	80.1377	77.8189
V_VM_VNG	359	587	388	639	92.5258	91.8623	92.7835	90.2973
TOTAL	86157	113113	88958	120859	96.8513	93.5909	94.9077	82.6393

Table. 15. Accuracy Rate per POS Category of the SVM Tagger (1= evaluation is done with the seen data, 2= evaluation is done with unseen data)

4.6.2.1.3.2.Accuracy per POS Category of the CRF++ Tagger

The following data is the representation of the accuracy per parts of speech category for the Odia CRF++ statistical tagger with precision and recall. As has already been observed in the SVM evaluation, the similar sorts of errors are committed by this tagger as well. Recall neither in the seen nor in the unseen domain has figured 100% mark of accuracy.

In the domain of seen data, the categories like echowords, punctuations, and reciprocal pronouns have registered cent percent accuracy in terms of precision while indefinite and interrogative pronouns including interrogative demonstrative have registered the lowest accuracy rates. The categories like reflexive pronoun, finite and auxiliary verbs, coordinator, symbol, cardinals, and classifiers have more than 98% accuracy. As far as recall is concerned, categories like common noun, reflexive pronoun, default particle, negative, symbol, and punctuation have registered more than 98% accuracy whereas the lowest rate of accuracy rates has been observed in foreign words, personal, reciprocal, and interrogative pronouns.

So far as the precision in the unseen data is concerned, pronouns (reciprocal and indefinite) and demonstrative (relative and interrogative) have registered 100% accuracy while common nouns, unknown words, and interrogative pronoun have the lowest accuracies. The categories like coordinators, reflexives, punctuations, deictic demonstrative, default particle, infinitive, and gerundive verbs have registered more than 98% rate of accuracy. So far as the recall is concerned, the categories like punctuation, symbol, negation, default particle, and coordinator have more than 98% accuracy while common and verbal nouns, reciprocal pronoun, and unknown words have the lowest accuracy rates.

Accuracy per Part-of-Speech for the Odia CRF++ Tagger as Precision and Recall

		Seen data		**Unseen data**	
Id	**Tag**	Recall	Precision	Recall	Precision
1	N_NN	98.709335	90.25096	96.4324	79.78141
2	N_NNP	81.48889	95.09855	54.30813	80.87242
3	N_NST	93.92838	96.327835	87.709496	97.616585
4	N_NNV	29.87805	94.230774	47.435898	97.368416
5	PR_PRP	97.35269	93.13725	96.70502	97.31579
6	PR_PRF	98.39572	99.45946	94.09091	99.75903
7	PR_PRL	74.46808	94.59459	87.36842	97.64706
8	PR_PRC	50	100	61.538464	100
9	PR_PRQ	28.813559	70.83333	92.85714	76.47059
10	PR_PRI	81.818184	45	87.03704	100

11	DM_DMD	94.18472	97.90596	96.60574	99.79023
12	DM_DMR	92.14559	95.62624	93.69748	100
13	DM_DMQ	68.42105	67.70833	75.59524	100
14	DM_DMI	93.36283	97.53467	94.5746	98.333336
15	V_VM	77.21519	91.27182	72.07729	89.77136
16	V_VM_VF	98.22385	98.235794	94.86219	93.51035
17	V_VM_VNF	69.366196	90.78341	73.998764	93.028656
18	V_VM_VINF	86.82745	91.540344	79.01419	99.34272
19	V_VM_VNG	70.62147	97.15026	71.830986	98.28694
20	V_VAUX	90.68965	98.50188	85.454544	97.91667
21	JJ	89.96068	95.48715	68.19492	90.69411
22	RB	82.84543	92.78688	80.28455	89.874855
23	PSP	96.03387	97.26923	93.91563	97.39204
24	CC_CCD	95.168	98.087074	98.40263	98.35276
25	CC_CCS	86.76868	90.34676	96.823296	96.69531
26	RP_RPD	99.075905	97.089264	98.95052	98.06835
27	RP_INJ	44.444447	88.88889	82.35294	100
28	RP_INTF	91.735535	95.68965	80.80169	91.40812
29	RP_NEG	98.73684	93.987976	99.10113	99.66102
30	RD_RDF	69.76744	93.75	0	0
31	RD_SYM	99.50739	98.29684	99.633026	99.633026
32	RD_PUNC	99.82036	100	99.579605	99.99308
33	RD_UNK	Gold data not found	0	17.509727	27.60736
34	RD_ECH	5.263158	100	0	0
35	QT_QTF	93.69104	92.43746	89.959015	95.95628
36	QT_QTC	88.37971	98.66017	77.00499	97.63713
37	QT_QTO	89.26553	95.757576	85.68966	99.4
38	RP_CL	81.8408	98.7988	69.047615	99.754295
Overall		94.39		88.87	

Table. 16. Accuracy Rate per POS Category of the CRF++ Tagger (1= evaluation is done with the seen data, 2= evaluation is done with unseen data)

4.6.2.2.Human Evaluation

Human evaluation is one of the most important approaches for looking at the errors committed by a statistical machine to evaluate the qualitative nature of the data at hand.

4.6.2.3.Evaluation through Inter Annotator Agreement

It is quite prominent that along with the automated evaluation guided by the quantitative perspective of the research, a computer application also needs to be evaluated from the qualitative approach to check the reliability of the data.

The results of the inter-annotator agreement are based on the data of around 3,000 (more exactly 2949 tagger outputs) and three Odia native linguists' annotation judgment of the language. The overall accuracy was calculated by the simple average formulae.

Average judgment in accuracy = ANN 1 + ANN 2 + ANN 3 / number of annotators

INTER ANNOTATOR AGREEMENT						
models	**SVM Evaluation**			**CRF++ Evaluation**		
annotators	**ANN 1**	**ANN 2**	**ANN 3**	**ANN 1**	**ANN 2**	**ANN 3**
accuracy	93.78	93.99	93.9	91.34	90.61	90.9
average accuracy	93.89%			90.95%		
all agree	92.21%			90.44%		
all disagree	4.79%			8.56%		

Table. 17. The Inter Annotator Agreement Report

The tabulated data demonstrates the fact that the average accuracy of the SVM IA judgment is 93.89% while CRF++ has 90.95%. Furthermore, the total accuracy of the tokens where all the annotators have agreed with a consensus is 92.21 and 90.44 respectively for both SVM and CRF++ models. The cases where all of them have disagreed account for 4.79 percent and 8.56 percent respectively. The cases of parts of speech where the annotators have largely disagreed are common nouns, adjectives, proper, coordinating and subordinating conjunctions, and deictic and indefinite demonstratives. This could be accounted for the fact that there are ambiguity issues, multiword expressions, foreign and unknown words, difficult linguistics, the gapping in the lexicon and so on. Ambiguity has been one of the major concerns for the natural language processing and they are of many types. They are lexical, semantic, syntactic, verbal, discourse and so on. One of the ways to deal with the issue is to tag correctly all the words and by maximizing the number of the types tokens where the taggers are committing errors. However, resolving all the ambiguities is just next to impossible, although lexical ambiguity can be handled.

4.6.3. Error Analysis

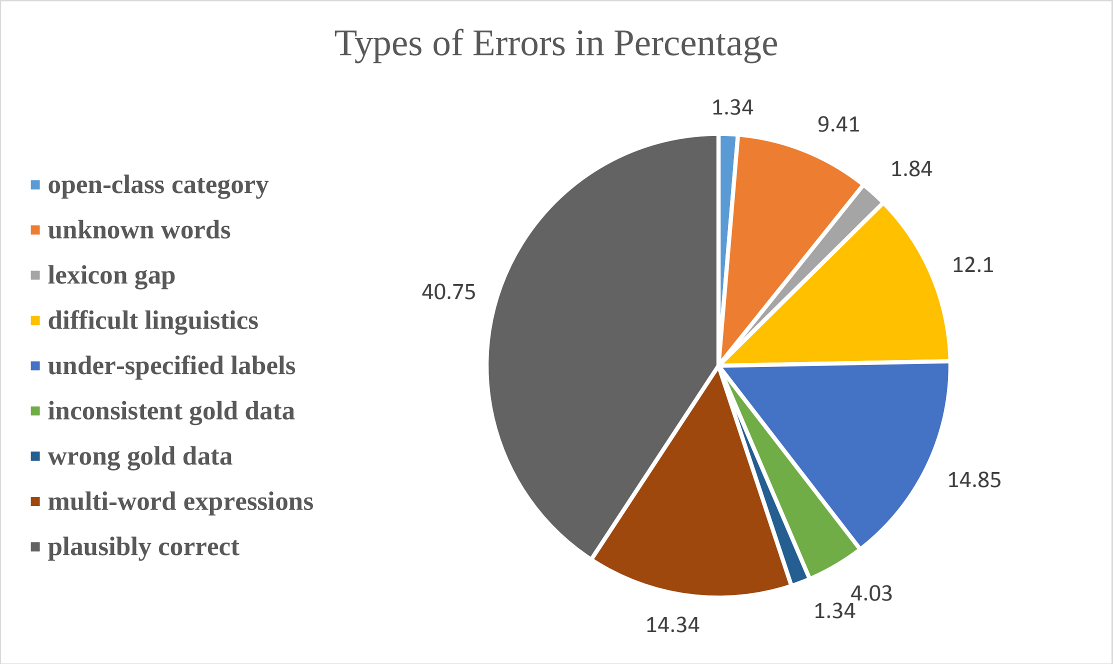


Chart. 2. The Overall Error Rate in Percentage

Some of the terminologies have been taken from Manning (2011). To increase the efficiency of the taggers in terms of the accuracy, one needs to ponder over the errors that the taggers commit. While identifying errors, it has been found out that there are some overlapping types of errors. The Pie chart above demonstrates the categories of errors and where one really should look at. The errors from both the taggers of around 250 (from an output of 3k data from both the taggers) have been categorized under 10 broad categories. They are as follows.

4.6.3.1.Open-class Categories

There are two basic categories of words or parts of speech: open-class categories and close-class categories. The open-class categories are words that can change over time or which can add new or loan words to its lexicon. On the other hand, the close class are the parts of speech in a language that are fixed in nature and there is little or no scope for the expansion of the lexicon. The examples of open-class are nouns, verbs and adjectives while the close class are the fixed function words such as prepositions or postpositions, adverbs, demonstratives, pronouns, conjunctions, particles, determiners, symbols, punctuations etc. The open-class foreign words can add to the lexicon by enculturation, getting transliterated or used Odia-like.

For example

Erɪɟenkɪs\N_NN (‘Erizenkis’ a proper noun from English)

kɔnʈɪle\N_NN bɔɳɔ\N_NN (the premodifier is an adjective in Hindi)

4.6.3.2.Unknown Words

The unknown words are the words that do not appear for even a single occurrence in the training corpus. These words could be from the same language or transliterated into Odia from other languages that are foreign. The known words have already registered their presence at least for once in the training data.

For example

Sɪsʈɔm\N_NNP ɔpʰ\N_NNP ɪnʈensɪpʰɪkesɔn\N_NNP ‘System of Intensification’

This is an example of a transliterated multi-word expressions from English which never occurred in the training data.

4.6.3.3.Lexicon Gap

If a word has occurred several times in the training data with a specific tag, but when it is evaluated, it gets a different tag by the tagger. These types of errors are more by the CRF++ tagger than the SVM.

For example

Training data token

mɒʈ\QT_QTF ɑɭʊrɔ\N_NN pɑkʰɑpɑkʰɪ\RB pɔnd̪ɔrɔ\QT_QTC prɔt̪ɪsɔt̪ɔ\N_NN

Evaluation data token

mɒʈ\N_NN ɑɭʊrɔ\N_NN pɑkʰɑpɑkʰɪ\RB pɔnd̪ɔrɔ\QT_QTC prɔt̪ɪsɔt̪ɔ\N_NN

‘Around fifteen percent of the total potato’

In the above example, the only tag that the word /mɒʈ/ gets in all the cases is QT_QTF, but during evaluation, the wrong tag labeled by the tagger is common noun N_NN.

4.6.3.4.Difficult Linguistics

When some problematic and ambiguous tags are not even decidable by the human annotator correctly and the tagger labels it incorrectly. These types of cases pertain to the other disciplines of linguistics like syntax, semantics and discourse than morphology. In Odia, it is quite difficult to judge in the conjunct verb constructions (JJ/N_NN+V_VM) whether the first lexical component is noun or adjective in several cases. Similarly, the cases of adverbs, demonstratives etc. are the other cases.

For Example

mʊ̃\PR_PRP t̪ɑkʊ\PR_PRP nɪjɔnt̪rɪt̪ɔ\N_NN kɔlɪ\V_VM_VF

“I controlled him”

4.6.3.5.Under-specified Labels

Unclear, ambiguous or under-specified words are the words having more than one tags in the whole training corpus or contextually unclear or undeterminable. Ambiguous words can be of both the known and the unknown words. In Odia, the average ambiguity for the SVM tagger is 6.5408 tags per token in the unseen data and 1.7482 tags per token in the unseen data. /mɑne/ is a word having three tags which create ambiguity during the processing of evaluation data. Some words have other linguistic ambiguities like lexical, syntactic, semantic etc.

For Example

/mɑne/ (CC_CCD or N_NN or PSP) ‘meaning’

4.6.3.6.Inconsistent Gold Standard

There are some cases where it becomes quite difficult to take proper judgment and the annotators disagree to arrive at a mutual consensus. As a result of the disagreement, they annotate some words based on their linguistic knowledge and thereby making the data inconsistent. Because of the inconsistency of both gold training annotated data, the evaluated data becomes error-prone. In the data explained below, /hɔɟɑrɔ/ and /sɔhɔ/ in both the training and gold file have been tagged inconsistently which results in an inconsistent output.

For Example

hɔɟɑrɔ\QT_QTF hɔɟɑrɔ\QT_QTC lɒkɔ\N_NN ‘thousands of people’

sɔhɔ\QT_QTF sɔhɔ\QT_QTC lɒkɔ\N_NN ‘hundreds of people’

4.6.3.7.Wrong Gold Data

When the data in the gold file has been annotated wrongly, the evaluated data also becomes wrong. For example, the multi-word in the gold data has been annotated wrongly; thereby making the evaluated data wrong invariably.

For example

ɔnt̪ɔrrɑsʈrijɔ\JJ ɑɭʊ\N_NN ɔnʊsɔnd̪ʰɑnɔ\N_NN kend̪rɔrɔ\N_NN ‘International Potato Research Centre’

4.6.3.8.Multi-word Expressions

This is one of the most prominent issues that has been much discussed in the parts of speech domain. There are a lot of compound proper nouns that occur as multi-words. Even human annotator is really confounded as to how to annotate the other parts of the multi-words except the proper noun. Because multi-word proper nouns contain some other elements that may not be proper generally; having their respective general tags. There could be some modifiers appearing before the proper noun in a multi-word. Therefore, as represented in the chart above, this category has the largest frequency of errors. In the following example, the word /sɔnɟʊkt̪ɔ/ occurs as an adjective mostly. Therefore, this word in the following multi-word has been tagged wrongly as the adjective and the following word as a common noun that should be part of a proper named entity.

For Example

sɔnɟʊkt̪ɔ\JJ rɑsʈrɔ\N_NN ɑmerɪkɑ\N_NNP ‘United States of America’

4.6.3.9.Plausibly Correct

There are some cases that suggest that even if there is correctness in both the training corpus and the gold file, they are tagged quite inconsistently by the taggers. These cases behave quite peculiarly; sometimes tagged rightly and sometimes wrongly. There are possibilities that these cases could be correctly assigned tags if features are selected taking into consideration the context information. The word /ɑgrɔhɔ/, which is a common noun, has been tagged as an adjective by the tagger.

For example

ɑgrɔhɔ\JJ srʊsʈɪ\N_NN kɔrɑɟɑɪcʰɪ\V_VM_VF ‘interest has been creared’

4.6.4. Suggested Solutions for the Statistical Taggers

This section contains different approaches: making the taggers hybrid by formulating linguistic rules, the data approach and words sense disambiguation, that have been proposed for the improvement of the performance in terms of the quality, reliability, and the efficiency of the statistical taggers. The only approach which has been applied and tested is the data approach. The other two approaches have been suggested and not attempted because of time constraints.

4.6.4.1.Formulation of Linguistic Rules

One of the methods for improving the performance of the tagger could be to formulate linguistic rules by observing the errors of both the statistical taggers. The encoding of these linguistic rules to the statistical taggers makes them hybrid in nature. On one hand, the present SVM tagger follows the learned information quite strictly and annotates the input data while the CRF++ tagger annotates uniquely. By ‘uniquely’, it refers to the fact that CRF++ tagger is based on the probability occurrences of the given input token to be evaluated. As a consequence, it annotates the input data considering its frequency of occurrences in the whole training data. By way of doing so, it selects the highest probable label as output for a given input token. For instance, if a token has two probable labels (N_NN and N_NNP): the former has 1234 times of occurrence in the whole training corpus while the latter has 1245 times of occurrence, it definitely selects the latter as it has the highest number of frequency in comparison to the former. This makes the CRF++ tagger performs less accurately in comparison to the SVM model. To increase the performance, a hybrid approach has been proposed. The hybrid approach will be an amalgam of both the statistical (probability-based or classifier-based approaches) and the contextual linguistic rules-driven. Some of the rules have been proposed taking the context features of the following and the preceding tags or tokens into accounts. They are as follows:

- Whenever spatio-temporal nouns (having the tag of N_NST) carry the genitive marker /-rɔ/ they are to be annotated as adjectives (JJ).
- When commonly-used general quantifiers are used before adjectives, adverbs, and quantifiers, they are tagged as intensifiers. Otherwise, when they precede nouns, they are general quantifiers.
- When /ɑʊ/ and /ɑhʊrɪ/ are used as coordinators, coordinating words, phrases and clauses, are tagged as coordinators. When they are used as prenominal modifiers, they are tagged as general quantifiers.
- When /ɔt̪ɪrɪkt̪ɔ/ precedes a noun phrase, it needs to be tagged as an adjective. When it follows a noun phrase, it can be tagged as a postposition.
- Whenever the word "/t̪ɔ/ is preceded by conjunct words" (Nainwani, 2012), it can be annotated as a conjunct. Otherwise, it is a particle by default.
- When /bʰɑbɔre/ is preceded by an adjective, it is an adverb. Or else, it is a common noun.
- When the word /ɟe/ is used as the complementizer augmenting a following subordinate phrase, it is tagged as a subordinating conjunction.
- When /pɑkʰɑ pɑkʰɪ/ occurs before a prenominal cardinal, it is tagged as an adverb since it is used in the sense of 'approximately'. If it is used as a modifier to noun just preceding it, it has been tagged as an adjective.

If these above discussed rules are encoded to the taggers, the accuracy rate of the statistical taggers in terms of accuracy, quality and reliability could be increased.

4.6.4.2.The Data Approach

The accuracy rate stage-wise shows that with the increase in the number of the tokens, the accuracy rate of the tagger increases. With each evaluation, results were evaluated and error analysis has been conducted manually. Based on the rule judgments of the human evaluator, corrections have been made. Initially, the accuracy rate has been evaluated manually, but the final evaluation has been performed by the machine. At the first stage with a training data of around 56k tokens the rate of accuracy was around 82%, with 86k it was 86%, with 113k it was 93% and with 130k it rose up to 96.85%. But, when it has been tested with the unseen data, the accuracy decreases to 93.59 because of a number of unknown and ambiguous words found by the taggers.

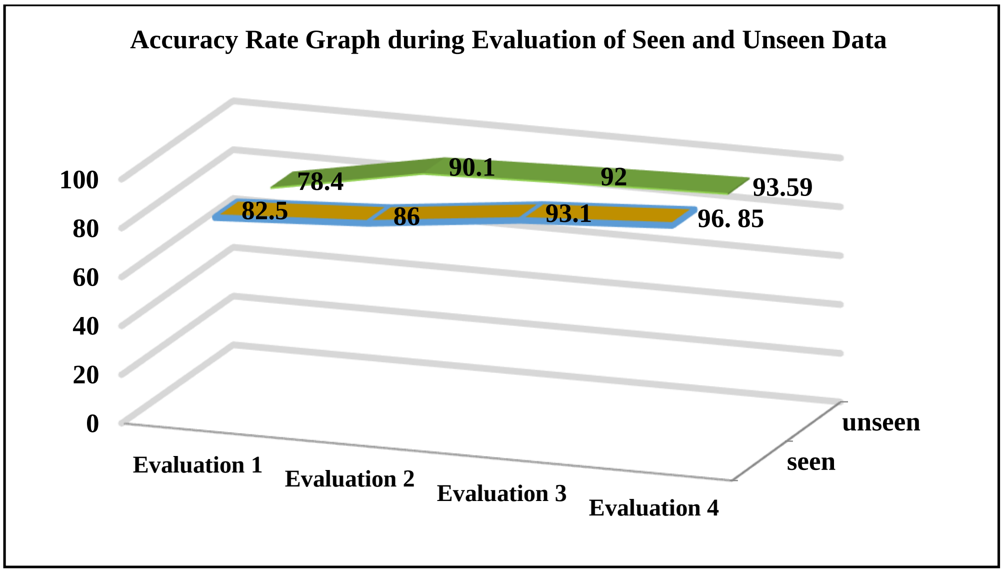


Chart. 3. Development of the Accuracy Rate during the Evaluation Period of the Seen and Unseen Data

4.6.4.3.Problematic Parts of Speech and Word Sense Disambiguation

It is often quite difficult to decide as to which annotation label is best suitable for a particular word even within a given context. When there is ambiguity or confusion, the context along with the native speakers' linguistic knowledge has been given utmost importance for deciding the tag of a given word. This section discusses the complicated annotation decisions taken while tagging the whole corpus. The first sub-section presents the parts of speech that 'can easily be confused and instructions on how to annotate' (Santorini, 1990) such sorts of cases if faced during annotation. The second sub-section has an alphabetical list of some problematic words and collocations.

4.6.4.3.1. Problematic Cases of POS and Disambiguation

"Categorial ambiguity arises when a particular word form can, in different instances, represent different grammatical categories" (De Rose, 1990). The ambiguity also arises when a particular word form has different tags at the same kind of contexts. This sub-section presents the parts of speech that can easily be confused and instructions on how to tag such sorts of cases. Further, it is noteworthy to mention that in this section only the lexical ambiguities (token-wise and label-wise) have been addressed.

- **CC_CCD or QT_QTF**

When /ɑʊ/ and /ɑhʊrɪ/ are used as coordinators, coordinating words, phrases, and clauses, they are tagged as coordinators.

Example mõ ɑʊ\CC_CCD mɒ b^{h}ɑɪ ‘I and my brother’

When they are used as prenominal modifiers, they are tagged as general quantifiers.

Examples ɑʊ\QT_QTF ekɔ ‘another one’

ɑhʊrɪ\QT_QTF d̪ʊɪʈɪ ‘another two’

- **DM_DMI or DM_DMQ**

When the demonstratives are used to refer to indefinite things, persons etc. in a declarative sentence, they are tagged as indefinite demonstrative.

Examples keʊ̃\DM_DMI kɑmɔ t̪ɑkʊ mɔ ɔɟɔɳɑ ‘which work is unknown to him.’

kɔʊɳɔsɪ bɪ kɑmɔ hɔʊ mõ kɔrɪd̪ebɪ ‘be it any work, I will do’

Analogously, if the same words are used for interrogation, they are tagged as the question/interrogative demonstratives.

Examples keʊ̃\DM_DMQ kɑmɔ kɔt̪^{h}ɑ kɔhʊc^{h}ɔ? ‘Which work are you talking about?’

t̪ɔme kɔʊɳɔsɪ\DM_DMQ bɪ kɑmɔ kɔrɪ ɟɑɳɪc^{h}ɔ? ‘Do you know about any work to do?’

- **DM_DMR or PR_PRL**

In the first example below, the referent of the relative demonstrative is present just following the relativizer. In these cases, the constructions have been tagged as relative demonstratives. But, when the referent is not present, relativizer has been tagged as a relative pronoun as in the second example.

Examples ɟe\DM_DMR lɒkɔ\N_NN cɪt̪t̪ɔ nɪrmɔɭɔ kɔrɪ kɑmɑd̪ɪre ɑsɔkt̪ɔ hʊe

‘that person who is attracted towards physical pleasures purifying soul’

ɟe\PR_PRL cɪt̪t̪ɔkʊ bɔsib^{h}ʊt̪ɔ kɔrɪbɑre sɔmɔrt̪^{h}ɔ ‘that person who keeps the soul under control’

- **N_NST or JJ**

When spatio-temporal nouns are used with or without genitive markers modifying the following nouns or noun phrases, they are tagged as adjectives. Otherwise, they are tagged as spatio-temporal nouns.

With genitive case markers at the attributive position:

Examples eʈʰɑ-rɔ\JJ gʰɔʈɔɳɑ\N_NN ‘incident of this place’

seʈʰɑ-rɔ\JJ lɒkɔ\N_NN ‘people of that place’
purb-ɔrɔ\JJ kɔt̪ʰɑ\N_NN ‘previous matter’

Without genitive case markers at the attributive position:

Examples purbɔ\JJ kɔt̪ʰɑ\N_NN ‘previous matter’

t̪ɔɭɔ\JJ kɒʈʰɔrɪ\N_NN ‘ground room’

When they are used with reduplicative and agglutinative forms, they are used as spatio-temporal nouns.

Examples purbɔrʊ\N_NST ‘from before’

pɔcʰe-pɔcʰe\N_NST ‘behind’ (in the sense of following)

- **N_NNP or JJ**

The adjectives in compound or complex proper nouns are tagged as proper nouns as they are the parts and partials of the proper noun. When the same word occurs as pre- or post-modifiers to common nouns, they are tagged as adjectives.

Examples rɑsʈrɪjɔ\N_NNP krʊsɔkɔ\N_NNP srɔmɪkɔ\N_NNP sɔŋɔʈʰɔnɔ\N_NNP ‘National Farmers’ Labour Union’

rɑsʈrɪjɔ\JJ git̪ɔ\N_NN ‘national song’

- **N_NN or V_VM_VF**

When the commonly-used verbs are used as nouns, they are tagged as nouns. Otherwise, they are tagged based on the context they are in.

Examples kɔhɔnt̪ɪ kɔre\N_NNP kɔrɔ\N_NN d̪ʰɔrɪ “tells Kara having held the hand.”

t̪ɔme ehɪ kɑmɔkʊ kɔrɔ\V_VM_VF “you just do this work.”

- **QT_QTF or RP_INTF**

When commonly-used general quantifiers are used before adjectives, adverbs, and quantifiers, they are tagged as intensifiers. Otherwise, when they precede nouns, they are general quantifiers.

Examples ɔɭpɔ\RP_INTF sʊɔt̪ɔnt̪rɔ\JJ ‘less free’

ʈɪke\RP_INTF sɑbɔd̪ʰɑnɔt̪ɑrɔ\RB sɔhɔ\RB ‘little carefully’

kʰʊb\RP_INTF ɟɒrre\RB ‘very loudly’

kʰʊb\QT_QTF pɔsɔnd̪ɔ\N_NN ‘much like’

- **QT_QTF or V_VM**

One has to really be cautious while dealing with such cases where one word form can be used with different tags having various functions. /kɔm/ can be used both as general quantifiers and as the main verb; the decision has been taken on the basis of the context.

Examples kɔm\QT_QTF kɔrɑɟɑʊcʰɪ ‘has been made less’

kɔm-ɑ\V_VM ɟɑʊɑcʰɪ ‘has been decreased’

- **QT_QTF or JJ**

One of the different cases in Odia is that the general quantifier like /ɔd̪ʰɪkɔ/ can have comparative suffixes like adjectives. However, these cases are not to be confused with the adjectives having comparative endings. They have been tagged as general quantifiers.

Examples ɔd̪ʰɪkɔ-t̪ɔrɔ\QT_QTF lɒkɔ ‘most of the people’

ɔd̪ʰɪkɔ-t̪ɔmɔ\QT_QTF kɑrɟjɔ ‘most of the work’

- **QT_QTF or QT_QTC**

When the numbers specified by the commonly-used cardinal quantifiers are uncountable or not exact, they are tagged as general quantifiers. If the number is fixed or countable, they have been tagged as cardinals.

Examples kɒʈɪ\QT_QTF kɒʈɪ\QT_QTF ʈɔnkɑ ‘crores of rupees’

sɔhɔ\QT_QTF sɔhɔ\QT_QTF gʰɔrɔ ‘hundreds of houses’

kɒʈɪe\QT_QTC ‘one crore’

sɔhe\QT_QTC ‘one hundred’

- **RB or JJ**

In the instances mentioned below, it has been decided that adjectives within the adverbial phrases have been tagged as adverbs.

Examples bʰɔlɔ\RB bʰɑbɔre kʰɑɔ ‘do it nicely’

ʈʰɪk\RB rupɔre kɔrɔ ‘do it well’

- **RB or N_NN**

In the instances below mentioned, it has been decided that the noun-like words followed by a postposition within an adverbial phrase have been tagged as an adverb.

Examples sɔpʰɔɭɔt̪ɑrɔ\RB sɔhɔ kɔrɔ ‘do it successfully’

gɔmbʰirɔt̪ɑrɔ\RB sɔhɔ ‘seriously’

ʊt̪sɑhɔ\RB sɔhɔ nɑcɔ ‘encouragingly’

- **RB or V_VM_VNF**

When the word /kɔrɪ/ occurs with the adjective /bɪsesɔ/, they function like adverbs modifying verbs directly or indirectly. They should not be confused with their canonical functions: /kɔrɪ/ as the non-finite verb and /bɪsesɔ/ as an adjective.

Examples sit̪ɔ rʊt̪ʊre bɔhut̪ɔ t̪ʰɔnɖɑ hʊe, bɪsesɔ\RB kɔrɪ\RB ʊt̪t̪ɔrɔ bʰɑgɔre

“It is quite cold in winter, especially in the northern part.”

- **V_VM or JJ**

When the noun-like components in a conjunct verb construction do neither function like verb (un-inflecting) nor do they function like noun, they have been tagged as adjectives. When they function like verbs in such cases, they have been tagged as verbs.

Examples bʰɔrt̪ɪ\JJ kɔrɔ\V_VM_VF ‘make it full’

bʰɔrt̪ɔ\V_VM_VF ‘full it’

sɔɭɔkʰɔ\JJ kɔrɪd̪ɪɔ\V_VM_VF ‘make it straight’

sɔɭɔkʰɑɔ\V_VM_VF ‘straighten it’

4.6.4.3.2. Annotating Cases of Specific Words and Collocations

- **/ɟe/: (CC_CCS or PR_PRL)**

When the word /ɟe/ is used as the complementizer augmenting a following subordinate phrase, it is tagged as a subordinating conjunction.

For instance kɪmbɔd̪ɔnt̪ɪ ehɑ mɔd̪ʰjɔ ɔʈe ɟe\CC_CCS eʈʰɑre hɔnʊmɑnɔnkʊ sɔnɟibɔni ɔʊsɔd̪ʰɔ mɪɭɪt̪ʰɪlɑ “the legend is that Hanuman had got Sanjeebani here.”

When /ɟe/ is used like relative pronoun relating to a preceding antecedent, it has been tagged as relative pronoun.

For instance ɟe\PR_PRL lɒkɔmɑnɔnkɔ mɔd̪ʰjɔre rɔhɪ ‘who, having stayed among people’

- **/ɟemit̪ɪ… semɪt̪ɪ/: (CC_CCS or PSP)**

When /ɟemit̪ɪ… semɪt̪ɪ/ are used as relative-correlative construction, they have been tagged as subordinating conjunctions.

Examples ɟemit̪ɪ ɟemit̪ɪ\CC_CCS

semɪt̪ɪ semɪt̪ɪ\CC_CCS

When they occur independently, they play the role of postpositions.

Examples ɟemit̪ɪ\PSP t̪ɔme kʰeɭɔ

semɪt̪ɪ\PSP kɔrɔ

- **/pɑkʰɑ pɑkʰɪ/: (RB or JJ)**

When it occurs before a prenominal cardinal, it is tagged as an adverb since it is used in the sense of ‘approximately’. If it is used as a modifier to noun just preceding it, it has been tagged as an adjective.

For Example pɑkʰɑ pɑkʰɪ\RB cɑrɪ\QT_QTC kɒʈɪ ʈɔnkɑ\N_NN ‘approximately 4 crores rupees’

mɒ pɑkʰɑ pɑkʰɪ\JJ gʰɔrɔ\N_NN ‘my nearest house’

- **/ɑd̪ɔʊ/: (QT_QTF or RP_INTF)**

When it is used for quantifying the quantity of nouns, it is a general quantifier. On the other hand, when it is used before adverbs or adjectives, it is tagged as intensifiers.

Examples ɑd̪ɔʊ\QT_QTF ɟɔɳɑnɑhĩ\V_VM_VF ‘does not at all know’

ɑd̪ɔʊ\RP_INTF kʰɔrɑpɔ\JJ ‘really bad’

/kɑrɔɳɔ/ (CC_CCD or N_NN or PSP)

If it is used after punctuations, verbs having all the sub-tags, symbols (exceptional cases) etc., it needs to be tagged as coordinating conjunctions. When it is used after adjectives, demonstratives, quantifiers, genitives etc., it needs to be tagged as common nouns. When the ablative case marker is attached or separated from it, it is tagged as postpositions.

Examples kɑrɔɳɔ\CC_CCD mʊ̃ ehɑ kɔrɪ pɑrɪbɪ nɑhĩ “because, I cannot do it.”

ehɪ spɔsʈɔ\JJ kɑrɔɳɔ\N_NN ɟɒgʊ̃ ‘because of this obvious reason’

ehɪ kɑrɔɳɔrʊ\PSP ‘because of this reason’

- **/mɑne/: (CC_CCD or N_NN or PSP)**

When /mɑne/ is used after punctuations, verbs and other coordinating conjunctions, it needs to be tagged as a coordinating conjunction. When it is used after adjectives, demonstratives, quantifiers, genitives etc., it needs to be tagged as common nouns. When it occurs just after the nouns and functions as the plural suffix, it needs to be tagged as postpositions. However, this ending can be part of different forms of pronouns and should not be confused with them.

Examples mɑne ehɪ ʈren ɟeʊ̃ʈʰɑrʊ bɑhɑre seɪʈʰɑre cʰɑɖɪnɔʈ̪ʰɑe “it means, this train does depart where it starts from”.

ehɑrɔ mɑne\N_NN ‘its mean’

lɒkɔ mɑne\PSP ‘people’

- **/d̪eɪ/: (PSP or V_VM_VNF)**

It can be both used as a postposition and a non-finite verb. When it is used after common and proper nouns, and postpositions, it is a postposition. However, it is not clear as to which occurrence has to be a postposition and non-finite verb as the difference is quite blurred since the selectional features apply to both the tags.

Examples pentɪ d̪eɪ\V_VM_VNF ɟɑɔ “go by giving me the pen”

ɟɔŋɔlɔ d̪eɪ\PSP ɟɑɔ nɑ “through the forest”

- **prɔt̪ɪ: (PSP or DM_DMI)**

This word can be both used as indefinite demonstrative and postposition. When it is used before nouns demonstrating indefiniteness, it has been tagged as an indefinite demonstrative. When it is used after a noun either as a case marker attached with the noun or as simple or complex postposition following the noun phrases or other such phrases, it is tagged as a postposition.

Examples prɔt̪ɪ\DM_DMI t̪ʰɔrɔ “every time”

ɔsɔhɑjɔnkɔ prɔt̪ɪ\PSP d̪ɔjɑ prɔd̪ɔrsɔnɔ kɔrɪbɑ ʊcɪt̪

“be kind towards the helpless”.

- **ɔt̪ɪrɪkt̪ɔ: (PSP or JJ)**

The concerned word can be used both as a postposition and an adjective. When it precedes a noun phrase, it has been tagged as an adjective. When it follows a noun phrase, it is tagged as a postposition.

Examples ɔt̪ɪrɪkt̪ɔ\JJ kʰɑd̪jɔ ‘surplus food’

ehɑ ɔt̪ɪrɪkt̪ɔ\PSP ‘besides this’

- **/ʊbʰɔjɔ/: (QT_QTF or PR_PRI)**

When /ʊbʰɔjɔ/ is used before common non-human nouns in the sense of both, it is tagged as the general quantifiers. When it is used to denote human nouns taking an oblique case /nkɔ/, it is tagged as an indefinite pronoun.

Examples ʊbʰɔjɔ\QT_QTF bɔhɪ\N_NN ‘both books’

ʊbʰɔjɔnkɔ\PR_PRI bɔhɪ\N_NN ‘books of both of them’

- **/sɔmɔst̪ɔnkɔʈʰɑrʊ/: (PR_PRI or RP_INTF)**

When /sɔmɔst̪ɔnkɔʈʰɑrʊ/ is used before adjectives, they are tagged either as an indefinite pronoun or an intensifier.

Examples sɔmɔst̪ɔnkɔʈʰɑrʊ\PR_PRI/RP_INTF bɔɖʰɪɑ\JJ ‘better of all’

sɔmɔst̪ɔnkɔʈʰɑrʊ\PR_PRI/RP_INTF kʰɔrɑpɔ\JJ ‘worse of all’

- **/sɔbʊʈʰɑrʊ/: (RP_INTF or PR_PRI)**

When /sɔbʊʈʰɑrʊ/ is used before adjectives directly, they are tagged as intensifier while if they are preceded by an intensifier, they are tagged as an indefinite pronoun.

Examples sɔbʊʈʰɑrʊ\RP_INTF bɔɖʰɪɑ\JJ ‘quite beautiful of all’

sɔbʊʈʰɑrʊ\PR_PRI ɔd̪ʰɪkɔ\RP_INTF bɔɖʰɪɑ\JJ ‘most beautiful of all’

- **/ʊcɪt̪/: (V_VAUX or JJ)**

When it is used in an attributive position, it is tagged as an adjective. When it is used as a modal auxiliary, it is tagged as an auxiliary verb.

Examples t̪ɔme kʰɑɪbɑ ʊcɪt̪\V_VAUX “you should eat”

ʊcɪt̪\JJ ɟɪnɪsɔ kʰɑɪbɑ kɔt̪ʰɑ “you need to take right kind of food”

- **/kebekebe/: (RB or DM_DMQ)**

When /kebekebe/ is used in the sense of ‘sometimes’ /beɭe beɭe/, it is an adverb. When /kebe/ is used as an interrogative for asking a question, it is an interrogative demonstrative.

Examples mɔ̃ kebekebe\RB ɟɑe “I go sometimes”

kebe\DM_DMQ t̪ʊme ɟɑɪcʰɔ ? “Have you ever been?”

- **/prɔt̪ʰɔmɔ/: (RB or N_NST or QT_QTO)**

This word form appears to be quite simple and its tag easily decidable. But, in fact, it is not so. When any slight changes come about in the root form of the word, it becomes inherent that its tag is going to be different. This word form forms the ambiguity sets with adverb, spatio-temporal nouns and ordinal quantifier.

Examples prɔt̪ʰɔmɔt̪ɔʔ\RB ‘firstly’

prɔt̪ʰɔmɔre\N_NST ‘at the first place’

prɔt̪ʰɔme\QT_QTO ‘first’

- **/bɑhɑrɔ/: (V_VM or N_NST or JJ)**

When this word form is used as a verb after the object, it is either a main or the finite form of the verb. When it is used as referring to a place with the addition of locative case marker, it is a spatio-temporal noun. When it is used in an attributive position, it is tagged as an adjective.

Examples bjɑgʈɪ bɑhɑrɔ\V_VM kɔrɔ “Open the bag.”

gʰɔrʊ bɑhɑrɔ\V_VM_VF “come out of the house.”

bɑhɑrɔkʊ\N_NST cɑlɔ “go out.”

bɑhɑrɔ\JJ gʰɔrɔkʊ cɑlɔ “go out to the outside house”

- **/ʈʰɑrʊ/: (N_NN or PP)**

This word form is quite complicated, as it functions both as a common noun and postposition. However, many linguists will not agree with the fact that it is a common noun. When it is preceded by a demonstrative, it is quite clear that it functions like a noun. On the other hand, when it occurs preceded by a noun phrase, it functions like a postposition. The idea will be pretty much clear with the following examples.

Examples sehɪ ʈʰɑrʊ ‘from that place’

mɒ\PR_PRP ʈʰɑrʊ\PSP

CHAPTER 5

5. CONCLUSION

This is the concluding chapter of the undertaken study which is divided into four sections. The first section provides an overall summary of the whole research while the second one briefly demonstrates the results of the study. The following sections mention about the limitations of the current research and its implications.

5.1.An Overall Summary of the Research

In the Introductory chapter, the sub-sections like geographical distribution, prominent languages, historical development, and the script of Odia have been dealt with. In the second section, a review of Odia linguistic and computational linguistics scenario, in general, has been provided followed by POS research in Indian languages in particular. The following section deals with different machine learning approaches to parts of speech annotation, aims and objectives, research questions, hypothesis, rationale and introduction to the computational framework for Odia POS Tagger. In the following section of research methodology, method of corpora collection, salient linguistic features of the data, BIS Tagset for annotation under ILCI, Online, and semi-automated ILCIANN App v2.0 have been discussed under method of data collection. The method of data analysis explains the methodologies of training, testing, and evaluation of CRF++ and SVM statistical taggers.

The second chapter encapsulates two sections: a precise introduction to Odia morphology with respect to the grammatical categories and a brief sketch of the Odia syntax in terms of phrases. The first section deals with the description of the eleven parts of speech whereas the second section provides an introductory background to four different types of Odia phrases: noun, verb, adjective, and adverb.

The third chapter presents four prominent sections. The first one lays emphasis on providing a list of annotation labels used for the annotation of the corpora along with their corresponding abbreviated tags while the second one deals with the parts of speech description along with their corresponding labels. The following section contains the issues and challenges while annotating parts of speech using BIS tagset under the ILCI. The final section proposes solutions for annotation scheme.

The fourth chapter is governed by two significant approaches to research: the quantitative and the qualitative approaches. The fourth chapter of evaluation and analysis comprises of tagsets evaluation and statistical taggers evaluation. Under the tagsets evaluation, issues of designing tagsets, types of tagsets and their comparison (ILMT, MSRI, LDC-IL, BIS) have been discussed. The taggers output evaluation further consists of two sections viz. automated and human evaluation. Under automated evaluation, the accuracy of known vs unknown, ambiguous vs unambiguous, per level of ambiguity, per parts of speech of both seen and unseen data of SVM and CRF++ have been discussed vividly. The human evaluation has been conducted on the basis of disagreement of Inter Annotator Agreement report carried out on 3k tokens tagger output. A short evaluation summary, error analysis, a list of parts of speech (tag-wise and word-wise), and words sense disambiguation have been provided finally.

In the concluding section, a summary, limitations of this research and further scope, and future research have been presented.

5.2.Results of the Study

The overall results obtained from the two statistical taggers: CRF++ (94.39 and 88.87) and SVM (96.85 and 93.59), demonstrate the fact that the latter performs better than the former with differences of 2.46 and 4.72 percentage in both the seen and unseen data respectively. Firstly, the under-performance of the CRF++ tagger can be ascribed to its probability method whereas for the competitive functioning of the SVM tagger, its context-based classifying feature can be held responsible. Secondly, because of the huge volume of the data to the SVM model, it performs efficiently while CRF++ does not solely depend on the data. To put forth, in other words, CRF++ needs the encoding of better features selection considering the salient linguistic features of the given language and is capable of providing competitive results with a less number of data. For both the statistical models, the agglutinative feature of Odia language proves to be an obstacle in machine learning and for an increasing rate of accuracy.

5.3.Present Research and Limitations of the Statistical Taggers

There are several limitations of the statistical taggers in the present undertaken research. One of the main objectives of the research has been to experiment statistical models for the Odia corpora collected under the ILCI Project. Another objective has been to be able to arrive at a conclusion as to which model is best suitable for the Odia corpora.

This is the research output after six-fold validation of errors and customized training of the taggers without compilation of any other tools like NER, WSD, Morph Analyzer etc. along with the taggers. Furthermore, the inspite of many machine learning issues, the accuracy rate has been enhanced through data approach. During every phase of validation, errors have been collated based on the nature of which new customized data set has been encoded for training

In the undertaken research, it has been attempted to personally collect Odia corpora along the use of the ILCI data in the domain of literature and annotate them for this research. The rationale for including this corpora is that in this domain the rules of grammar, word order and so on are often violated which may have caused problems and resulted in lower accuracy rate, if one had evaluated the taggers with the literary data.

Another important limitation of the research is that simple features selections for both the statistical taggers have been made and based on which results have been evaluated. The features like Unigram, verbose 1 etc. have been selected for the CRF++ tagger while the features like medium verbose (-V 2) and left-right-left (LRL) mode have been used for the annotation process. The rest of the features for both the taggers have been selected to default mode.

So far as the data is concerned, it is a conglomeration of five domains viz. health, tourism, agriculture, entertainment, and literature. In the corpora, the data for verbal noun and echo words are not so exhaustive. As already discussed, the case markers are attached with almost all the categories like demonstratives, pronouns, general quantifiers, nouns, verbs etc. in an agglutinating form which creates lexical ambiguity for the taggers. However, this issue could be addressed applying a Morph Analyzer, contextual disambiguator and selecting better contextual features for the words and parts of speech labels.

5.4.Scope and Implications of the Research

The present research will prove to be quite beneficial for further research and development in the field of Odia NLP. There are only two reported earlier research works on the statistical Parts of speech tagger for Odia: by applying the neural network technique and SVM. The neural network tagger reports a less accuracy of around 81% while the latter provides 82% in comparison to the present research output. This tool could be used for making chunker, parser, discourse anaphora resolution and machine translation platforms. Furthermore, 'semi-supervised learning approaches' can be applied to annotate automatically the freely available huge amount of unannotated data (Pathak et.al, 2014).

Patra et al., (2012) has discussed that the accuracy can be increased by the inclusion of 'lexicon' and 'inflection lists'. They have further observed that NER and MIS are really necessary to minimize the multi-words and proper nouns-related error-rate of the statistical taggers.

De Rose (1990) has observed that statistical annotation methods provide 'efficient means' of assigning parts of speech categories if the 'normalization corpora' are of 'adequate size'. Generally, the statistical tagger functions robustly provided all its required criteria are met. The performance of the tagger decreases only when 'adequately normalized'. A practical dictionary word forms of 35, 000 - 75,000 volume should provide more than 90 percent accuracy which may be applied for increasing the performance and disambiguation purposes. The next step can be to handle the unknown words so that the accuracy of the Odia SVM tagger can be ensured.

Because of the extremely high degree of grammatical category ambiguity in natural language, "NLP systems come to terms with excessive non-determinism".[69] Therefore, if one applies better feature selection taking the Odia linguistic features into consideration, the accuracy rate could be increased by disambiguation. In addition, one can also apply tools like WSD, NER, Morph Analyzer, a suitable tokenizer, lexical database with prefix and suffix, dictionary look-up and post-processor to increase the accuracy rate of the taggers. For the application of the WSD, the list of the problematic cases has already been provided in the evaluation section. Besides, if one does have a look on the types of accuracy and works on correcting them, then also the accuracy can be enhanced.

[69] Ibid.

De Rose has stated that probabilistic methods are best suitable for NLP. Considering the present study, it can be averred that the SVM model works best for the Odia ILCI corpora. Because, in both the domains of seen and unseen data, the accuracy of the SVM model outperforms the CRF++ tagger.

APPENDICES

Appendix I

Representative Inter Annotator Agreement Data for SVM and CRF Taggers

Sl.		SVM Tagger				CRF Tagger				
2	INPUT TOKENS	TAGS	ANN 1	ANN 2	ANN 3	TAGS	ANN 1	ANN 2	ANN 3	COMMENTS
3	୧୯୭୪	QT_QTC	y	Y	y	QT_QTC	y	y	y	
4	ମସିହାରେ	N_NN	y	Y	y	N_NN	y	y	y	
5	ଏରିକଜେଙ୍କିସ୍	N_NN	n	N	n	N_NN	n	n	n	open and close class
6	ଗୋଟିଏ	RP_CL	y	Y	y	RP_CL	y	y	y	
7	ଗଛରୁ	N_NN	y	Y	y	N_NN	y	y	y	
8	୧୬୮	QT_QTC	y	Y	y	QT_QTC	y	y	y	
9	କିଲୋଗ୍ରାମ	N_NN	y	Y	y	N_NN	y	y	y	
10	ଆଳୁ	N_NN	y	Y	y	N_NN	y	y	y	
11	ଉତ୍ପାଦନ	N_NN	y	Y	y	N_NN	y	y	y	
12	କରି	V_VM_VNF	y	Y	y	V_VM_VNF	y	y	y	
13	ବିଶ୍ୱରେକର୍ଡ	N_NN	y	Y	y	N_NN	y	y	y	
14	ସୃଷ୍ଟିକରିଥି୍ଲେ	V_VM_VF	y	y	y	V_VM_VF	y	y	y	
15	ଯାହା	DM_DMR	y	y	y	DM_DMR	y	y	y	
16	ଆଜି	RB	y	y	y	RB	y	y	y	
17	ମଧ୍ୟ	RP_RPD	y	y	y	RP_RPD	y	y	y	
18	ବଳବତ୍ତର	N_NN	n	n	n	N_NN	n	n	n	open and close class
19	ଅଛି	V_VM_VF	y	y	y	V_VM_VF	y	y	y	
20	।	RD_PUNC	y	y	y	RD_PUNC	y	y	y	
21			y	y	y		y	y	y	
22	ଆଳୁର	N_NN	y	y	y	N_NN	y	y	y	
23	କିଛି	QT_QTF	y	y	y	QT_QTF	y	y	y	
24	କିସମରେ	N_NN	y	y	y	N_NN	y	y	y	
25	ବହୁତ	RP_INTF	y	y	y	RP_INTF	y	y	y	
26	ଉତ୍କଟ	JJ	y	y	y	JJ	y	y	y	
27	ଗନ୍ଧ	N_NN	y	y	y	N_NN	y	y	y	
28	ଆସିଥାଏ	V_VM_VF	y	y	y	V_VM_VF	y	y	y	

29	।	RD_PUNC	y	y	y	RD_PUNC	y	y	y	
30			y	y	y		y	y	y	
31	ସମଗ୍ର	JJ	y	y	y	JJ	y	y	y	
32	ୟୁରୋପରେ	N_NNP	y	y	y	N_NN	n	n	n	
33	ଆଳୁ	N_NNP	n	n	n	N_NN	y	n	n	multi-word
34	ମୁଖ୍ୟ	JJ	y	y	y	JJ	y	y	y	
35	ଭୋଜନ	N_NN	y	y	y	N_NN	y	y	y	
36	ଅଟେ	V_VM_VF	y	y	y	V_VM_VF	y	y	y	
37	।	RD_PUNC	y	y	y	RD_PUNC	y	y	y	
38			y	y	y		y	y	y	
39	ଭାରତର	N_NNP	y	y	y	N_NNP	y	y	y	
40	ସିମ୍ଲାରେ	N_NNP	y	y	y	N_NNP	y	y	y	
41	କେନ୍ଦ୍ରୀୟ	N_NNP	y	y	y	JJ	y	y	y	
42	ଆଳୁ	N_NNP	y	y	y	N_NN	n	n	n	
43	ସଂସ୍ଥାନ	N_NNP	y	y	y	N_NN	n	n	n	
44	କୁଫେରୀ	N_NNP	y	y	y	JJ	n	n	n	
45	ଶ୍ରେଣୀର	N_NN	y	y	y	N_NN	y	y	y	
46	ପାଖାପାଖି	RB	y	y	y	RB	y	y	y	
47	୪୫ଟି	RP_CL	y	y	y	N_NN	n	n	n	
48	କିସମ	N_NN	y	y	y	N_NN	y	y	y	
49	ବିକଶିତ	JJ	n	y	y	N_NN	y	n	n	difficult linguistics
50	କରି	V_VM_VNF	y	y	y	V_VM_VNF	y	y	y	
51	ଆଳୁ	N_NN	y	y	y	N_NN	y	y	y	
52	କ୍ରାନ୍ତିରେ	N_NN	y	y	y	N_NN	y	y	y	
53	ନିଜର	PR_PRF	y	y	y	PR_PRF	y	y	y	
54	ଭୂମିକା	N_NN	y	y	y	N_NN	y	y	y	
55	ନିର୍ବାହ	N_NN	y	y	y	N_NN	y	y	y	
56	କରିଛନ୍ତି	V_VM_VF	y	y	y	V_VM_VF	y	y	y	
57	।	RD_PUNC	y	y	y	RD_PUNC	y	y	y	
58			y	y	y		y	y	y	
59	ଆଳୁର	N_NN	y	y	y	N_NN	y	y	y	
60	ମହତ୍ତ୍ୱକୁ	N_NN	y	y	y	N_NN	y	y	y	
61	ଆଖୁ	N_NN	y	y	y	N_NN	y	y	y	
62	ଆଗରେ	N_NST	y	y	y	N_NST	y	y	y	
63	ରଖି	V_VM_VNF	y	y	y	V_VM_VNF	y	y	y	
64	ସଂଯୁକ୍ତ	JJ	y	y	y	N_NNP	y	y	y	
65	ରାଷ୍ଟ୍ର	N_NN	n	n	n	N_NNP	y	y	y	multi-word
66	ସଂଘ	N_NN	n	n	n	QT_QTC	n	n	n	multi-word
67	ବର୍ଷ	N_NN	y	y	y	N_NN	y	y	y	
68	୨୦୦୮କୁ	QT_QTC	y	y	y	N_NN	n	n	n	

69	‘	RD_PUNC	y	y	y	RD_PUNC	y	y	y	
70	ଆଳୁ	N_NNP	n	n	n	QT_QTC	n	n	n	under-specified
71	ବର୍ଷ	N_NN	y	y	y	N_NN	y	y	y	
72	’	RD_PUNC	y	y	y	RD_PUNC	y	y	y	
73	ଘୋଷିତ	JJ	n	y	y	N_NN	y	n	n	difficult linguistics
74	କରିଥିଲେ	V_VM_VF	y	y	y	V_VM_VF	y	y	y	
75	।	RD_PUNC	y	y	y	RD_PUNC	y	y	y	
76			y	y	y		y	y	y	
77	ଦୁନିଆରେ	N_NN	y	y	y	N_NN	y	y	y	
78	୧୨୫ଟି	RP_CL	y	y	y	N_NN	n	n	n	
79	ଦେଶରେ	N_NN	y	y	y	N_NN	y	y	y	
80	ଆଳୁର	N_NN	y	y	y	N_NN	y	y	y	
81	ଚାଷ	N_NN	y	y	y	N_NN	y	y	y	
82	କରାଯାଏ	V_VM_VF	y	y	y	V_VM_VF	y	y	y	
83	।	RD_PUNC	y	y	y	RD_PUNC	y	y	y	
84			y	y	y		y	y	y	
85	ଆଜି	RB	y	y	y	RB	y	y	y	
86	ବିଶ୍ୱରେ	N_NN	y	y	y	N_NN	y	y	y	
87	ଆଳୁର	N_NN	y			N_NN	y	y	y	
88	୫୦୦୦	QT_QTC	y	y	y	N_NN	n	n	n	
89	ପାଖାପାଖି	RB	y	y	y	RB	y	y	y	
90	କିସମ	N_NN	y	y	y	N_NN	y	y	y	
91	ଅଛି	V_VM_VF	y	y	y	V_VM_VF	y	y	y	
92	ଯାହା	DM_DMR	y	y	y	DM_DMR	y	y	y	
93	ସର୍ବାଧିକ	QT_QTF	y	y	y	RP_INTF	y	y	y	
94	ଆଣ୍ଡିଜ୍	N_NN	n	n	n	JJ	n	n	n	multi-word
95	ପାର୍ବତ୍ୟାଞ୍ଚଳରେ	N_NN	y	y	y	N_NN	y	y	y	
96	ଉତ୍ପାଦିତ	JJ	n	y	y	JJ	n	y	y	difficult linguistics
97	ହୋଇଥାଏ	V_VM_VF	y	y	y	V_VM_VF	y	y	y	
98	।	RD_PUNC	y	y	y	RD_PUNC	y	y	y	
99			y	y	y		y	y	y	

Appendix II

Representative Set of SVM Data Used for Training, Testing and Evaluation

SVM	Train Data	Gold Data	Tagger Output	Tokenized Input Text
1	ବ୍ୟାପାର N_NN	୧୯୭୪ QT_QTC	୧୯୭୪ QT_QTC	୧୯୭୪
2	ମାନ୍ଦାର N_NN	ମସିହାରେ N_NN	ମସିହାରେ N_NN	ମସିହାରେ
3	ଏହି DM_DMD	ଏରିକଜେଙ୍କିସ୍ N_NNP	ଏରିକଜେଙ୍କିସ୍ N_NN	ଏରିକଜେଙ୍କିସ୍
4	ଯୁଗରେ N_NN	ଗୋଟିଏ RP_CL	ଗୋଟିଏ RP_CL	ଗୋଟିଏ
5	ରାଜସ୍ଥାନର N_NNP	ଗଛରୁ N_NN	ଗଛରୁ N_NN	ଗଛରୁ
6	ଦୁଇ QT_QTC	୧୬୮ QT_QTC	୧୬୮ QT_QTC	୧୬୮
7	ରାଜକୀୟ JJ	କିଲୋଗ୍ରାମ N_NN	କିଲୋଗ୍ରାମ N_NN	କିଲୋଗ୍ରାମ
8	ରେଲଗାଡ଼ିଗୁଡ଼ିକ N_NN	ଆଳୁ N_NN	ଆଳୁ N_NN	ଆଳୁ
9	ପାଇଁ PSP	ଉତ୍ପାଦନ N_NN	ଉତ୍ପାଦନ N_NN	ଉତ୍ପାଦନ
10	ନୂଆଁ JJ	କରି V_VM_VNF	କରି V_VM_VNF	କରି
11	ପେକେଜ୍ N_NN	ବିଶ୍ୱରେକର୍ଡ N_NN	ବିଶ୍ୱରେକର୍ଡ N_NN	ବିଶ୍ୱରେକର୍ଡ
12	ଖୋଜା V_VM	ସୃଷ୍ଟିକରିଥିଲେ V_VM_VF	ସୃଷ୍ଟିକରିଥିଲେ V_VM_VF	ସୃଷ୍ଟିକରିଥିଲେ
13	ଯାଉଛି V_VM_VF	ଯାହା DM_DMR	ଯାହା DM_DMR	ଯାହା
14	ଏବଂ CC_CCD	ଆଜି RB	ଆଜି RB	ଆଜି
15	ଆସନ୍ତା JJ	ମଧ RP_RPD	ମଧ RP_RPD	ମଧ
16	ଟୁରିଷ୍ଟ JJ	ବଳବତ୍ତର JJ	ବଳବତ୍ତର N_NN	ବଳବତ୍ତର
17	ସିଜନରେ N_NN	ଅଛି V_VM_VF	ଅଛି V_VM_VF	ଅଛି
18	ହେଇପାରେ RB	। RD_PUNC	। RD_PUNC	।
19	ପେଲେସ୍ N_NNP			
20	ଅନ୍ N_NNP	ଆଳୁର N_NN	ଆଳୁର N_NN	ଆଳୁର
21	ହ୍ୱୀଲ୍ସରେ N_NNP	କିଛି QT_QTF	କିଛି QT_QTF	କିଛି
22	କୌଣସି DM_DMI	କିସମରେ N_NN	କିସମରେ N_NN	କିସମରେ
23	ଲିଜ N_NNP	ବହୁତ RP_INTF	ବହୁତ RP_INTF	ବହୁତ
24	ହର୍ଲେ N_NNP	ଉତ୍କଟ JJ	ଉତ୍କଟ JJ	ଉତ୍କଟ
25	କିମ୍ବା CC_CCD	ଗନ୍ଧ N_NN	ଗନ୍ଧ N_NN	ଗନ୍ଧ
26	ଅରୂଣ N_NNP	ଆସିଥାଏ V_VM_VF	ଆସିଥାଏ V_VM_VF	ଆସିଥାଏ
27	ନାୟାର N_NNP	। RD_PUNC	। RD_PUNC	।
28	ଭଳି PSP			
29	ସେଲେବ୍ରିଟି N_NN	ସମଗ୍ର JJ	ସମଗ୍ର JJ	ସମଗ୍ର
30	ନିଜର PR_PRF	ୟୁରୋପରେ N_NNP	ୟୁରୋପରେ N_NNP	ୟୁରୋପରେ
31	ବିବାହ N_NN	ଆଳୁ N_NN	ଆଳୁ N_NNP	ଆଳୁ
32	କରୁଥିବା V_VM_VNG	ମୁଖ୍ୟ JJ	ମୁଖ୍ୟ JJ	ମୁଖ୍ୟ
33	ଦେଖାଦେଇ V_VM	ଭୋଜନ N_NN	ଭୋଜନ N_NN	ଭୋଜନ
34	ପାରେ V_VM_VF	ଅଟେ V_VM_VF	ଅଟେ V_VM_VF	ଅଟେ
35	| RD_PUNC	। RD_PUNC	। RD_PUNC	।
36				

37	ଆଜ୍ଞା N_NN	ଭାରତର N_NNP	ଭାରତର N_NNP	ଭାରତର
38	ହଁ RP_INJ	ସିମ୍ଲାରେ N_NNP	ସିମ୍ଲାରେ N_NNP	ସିମ୍ଲାରେ
39	, RD_PUNC	କେନ୍ଦ୍ରୀୟ N_NNP	କେନ୍ଦ୍ରୀୟ N_NNP	କେନ୍ଦ୍ରୀୟ
40	ରେଳରେ N_NN	ଆଳୁ N_NNP	ଆଳୁ N_NNP	ଆଳୁ
41	ବିବାହ N_NN	ସଂସ୍ଥାନ N_NNP	ସଂସ୍ଥାନ N_NNP	ସଂସ୍ଥାନ
42	ଏବଂ CC_CCD	କୁଫେରୀ N_NNP	କୁଫେରୀ N_NNP	କୁଫେରୀ
43	ରୟାଲ N_NNP	ଶ୍ରେଣୀର N_NN	ଶ୍ରେଣୀର N_NN	ଶ୍ରେଣୀର
44	ରାଜସ୍ଥାନ N_NNP	ପାଖାପାଖି RB	ପାଖାପାଖି RB	ପାଖାପାଖି
45	ଅନ୍ N_NNP	୪୫ଟି RP_CL	୪୫ଟି RP_CL	୪୫ଟି
46	ହ୍ୱୀଲ୍ସରେ N_NNP	କିସମ N_NN	କିସମ N_NN	କିସମ
47	ମଧୁଚନ୍ଦ୍ରିକାର N_NN	ବିକଶିତ N_NN	ବିକଶିତ JJ	ବିକଶିତ
48	ପେକେଜ୍ N_NN	କରି V_VM_VNF	କରି V_VM_VNF	କରି
49	ଏହି DM_DMD	ଆଳୁ N_NN	ଆଳୁ N_NN	ଆଳୁ
50	ଗାଡ଼ିଗୁଡ଼ିକର N_NN	କ୍ରାନ୍ତିରେ N_NN	କ୍ରାନ୍ତିରେ N_NN	କ୍ରାନ୍ତିରେ
51	ସଞ୍ଚାଳନ N_NN	ନିଜର PR_PRF	ନିଜର PR_PRF	ନିଜର
52	ସମ୍ଭାଳୁଥିବା V_VM_VNG	ଭୂମିକା N_NN	ଭୂମିକା N_NN	ଭୂମିକା
53	ରାଜସ୍ଥାନ N_NNP	ନିର୍ବାହ N_NN	ନିର୍ବାହ N_NN	ନିର୍ବାହ
54	ପର୍ଯ୍ୟଟନ N_NNP	କରିଛନ୍ତି V_VM_VF	କରିଛନ୍ତି V_VM_VF	କରିଛନ୍ତି
55	ବିକାସ N_NNP	। RD_PUNC	। RD_PUNC	।
56	ନିଗମ N_NNP			
57	(RD_SYM	ଆଳୁର N_NN	ଆଳୁର N_NN	ଆଳୁର
58	ଆର୍ଟୀଡ଼ୀସୀ N_NNP	ମହତ୍ତ୍ୱକୁ N_NN	ମହତ୍ତ୍ୱକୁ N_NN	ମହତ୍ତ୍ୱକୁ
59	) RD_SYM	ଆଖୁ N_NN	ଆଖୁ N_NN	ଆଖୁ
60	ର PSP	ଆଗରେ N_NST	ଆଗରେ N_NST	ଆଗରେ
61	ଯୋଜନାଗୁଡ଼ିକରେ N_NN	ରଖି V_VM_VNF	ରଖି V_VM_VNF	ରଖି
62	ସାମିଲ N_NN	ସଂଯୁକ୍ତ N_NNP	ସଂଯୁକ୍ତ JJ	ସଂଯୁକ୍ତ
63	ଅଛି V_VM_VF	ରାଷ୍ଟ୍ର N_NNP	ରାଷ୍ଟ୍ର N_NN	ରାଷ୍ଟ୍ର
64	| RD_PUNC	ସଂଘ N_NNP	ସଂଘ N_NN	ସଂଘ
65		ବର୍ଷ N_NN	ବର୍ଷ N_NN	ବର୍ଷ
66	ଆର୍ଟୀଡ଼ୀସୀ N_NNP	୨୦୦୮କୁ QT_QTC	୨୦୦୮କୁ QT_QTC	୨୦୦୮କୁ
67	କାୟଦା N_NN	‘ RD_PUNC	‘ RD_PUNC	‘
68	ଅନୁସାରେ PSP	ଆଳୁ N_NN	ଆଳୁ N_NNP	ଆଳୁ
69	ନିଜର PR_PRF	ବର୍ଷ N_NN	ବର୍ଷ N_NN	ବର୍ଷ
70	ଏଜେଣ୍ଟମାନଙ୍କୁ N_NN	’ RD_PUNC	’ RD_PUNC	’
71	କହିଛନ୍ତି V_VM_VF	ଘୋଷିତ JJ	ଘୋଷିତ JJ	ଘୋଷିତ
72	ଯେ CC_CCS	କରିଥିଲେ V_VM_VF	କରିଥିଲେ V_VM_VF	କରିଥିଲେ
73	ସେମାନେ PR_PRP	। RD_PUNC	। RD_PUNC	।
74	ହଲିଉଡ୍କୁ N_NNP			
75	ଏହି DM_DMD	ଦୁନିଆରେ N_NN	ଦୁନିଆରେ N_NN	ଦୁନିଆରେ
76	ପେକେଜ୍ N_NN	୧୨୫ଟି RP_CL	୧୨୫ଟି RP_CL	୧୨୫ଟି
77	ପାଇଁ PSP	ଦେଶରେ N_NN	ଦେଶରେ N_NN	ଦେଶରେ

78	ଆଖିରେ N_NN	ଆଳୁର N_NN	ଆଳୁର N_NN	ଆଳୁର
79	ରଖନ୍ତୁ V_VM_VF	ଚାଷ N_NN	ଚାଷ N_NN	ଚାଷ
80	| RD_PUNC	କରାଯାଏ V_VM_VF	କରାଯାଏ V_VM_VF	କରାଯାଏ
81		। RD_PUNC	। RD_PUNC	।
82	ନୂଆଁ JJ			
83	ଯୋଜନାରେ N_NN	ଆଜି RB	ଆଜି RB	ଆଜି
84	ଆହୁରି QT_QTF	ବିଶ୍ୱରେ N_NN	ବିଶ୍ୱରେ N_NN	ବିଶ୍ୱରେ
85	ଗୋଟିଏ RP_CL	ଆଳୁର N_NN	ଆଳୁର N_NN	ଆଳୁର
86	ମୁଖ୍ୟ JJ	୫୦୦୦ QT_QTC	୫୦୦୦ QT_QTC	୫୦୦୦
87	କଥା N_NN	ପାଖାପାଖି RB	ପାଖାପାଖି RB	ପାଖାପାଖି
88	ଏହି DM_DMD	କିସମ N_NN	କିସମ N_NN	କିସମ
89	ରାଜକୀୟ JJ	ଅଛି V_VM_VF	ଅଛି V_VM_VF	ଅଛି
90	ଗାଡ଼ିଗୁଡ଼ିକର N_NN	ଯାହା DM_DMR	ଯାହା DM_DMR	ଯାହା
91	ଚାଲିବାର V_VM_VINF	ସର୍ବାଧିକ QT_QTF	ସର୍ବାଧିକ QT_QTF	ସର୍ବାଧିକ
92	ସମୟକୁ N_NN	ଆଣ୍ଡିଜ୍ N_NNP	ଆଣ୍ଡିଜ୍ N_NN	ଆଣ୍ଡିଜ୍
93	ମଧ୍ୟ RP_RPD	ପାର୍ବତ୍ୟାଞ୍ଚଳରେ N_NN	ପାର୍ବତ୍ୟାଞ୍ଚଳରେ N_NN	ପାର୍ବତ୍ୟାଞ୍ଚଳରେ
94	ବଢ଼ାଇବା V_VM_VINF	ଉତ୍ପାଦିତ N_NN	ଉତ୍ପାଦିତ JJ	ଉତ୍ପାଦିତ
95	ଅଟେ V_VM_VF	ହୋଇଥାଏ V_VM_VF	ହୋଇଥାଏ V_VM_VF	ହୋଇଥାଏ
96	| RD_PUNC	। RD_PUNC	। RD_PUNC	।
97				

Appendix III

Representative Set of CRF Data Used for Training, Testing and Evaluation

CRF	Train Data	Gold Data	Tagger Output	Tokenized Input Text
1	ବ୍ୟାପାର N_NN	୧୯୭୪ QT_QTC	୧୯୭୪ QT_QTC	୧୯୭୪
2	ମାନ୍ଦାର N_NN	ମସିହାରେ N_NN	ମସିହାରେ N_NN	ମସିହାରେ
3	ଏହି DM_DMD	ଏରିକଜେଙ୍କିସ୍ N_NNP	ଏରିକଜେଙ୍କିସ୍ N_NN	ଏରିକଜେଙ୍କିସ୍
4	ଯୁଗରେ N_NN	ଗୋଟିଏ RP_CL	ଗୋଟିଏ RP_CL	ଗୋଟିଏ
5	ରାଜସ୍ଥାନର N_NNP	ଗଛରୁ N_NN	ଗଛରୁ N_NN	ଗଛରୁ
6	ଦୁଇ QT_QTC	୧୬୮ QT_QTC	୧୬୮ QT_QTC	୧୬୮
7	ରାଜକୀୟ JJ	କିଲୋଗ୍ରାମ N_NN	କିଲୋଗ୍ରାମ N_NN	କିଲୋଗ୍ରାମ
8	ରେଳଗାଡ଼ିଗୁଡ଼ିକ N_NN	ଆଳୁ N_NN	ଆଳୁ N_NN	ଆଳୁ
9	ପାଇଁ PSP	ଉତ୍ପାଦନ N_NN	ଉତ୍ପାଦନ N_NN	ଉତ୍ପାଦନ
10	ନୂଆଁ JJ	କରି V_VM_VNF	କରି V_VM_VNF	କରି
11	ପେକେଜ୍ N_NN	ବିଶ୍ୱରେକର୍ଡ N_NN	ବିଶ୍ୱରେକର୍ଡ N_NN	ବିଶ୍ୱରେକର୍ଡ
12	ଖୋଜା V_VM	ସୃଷ୍ଟିକରିଥିଲେ V_VM_VF	ସୃଷ୍ଟିକରିଥିଲେ V_VM_VF	ସୃଷ୍ଟିକରିଥିଲେ
13	ଯାଉଛି V_VM_VF	ଯାହା DM_DMR	ଯାହା DM_DMR	ଯାହା
14	ଏବଂ CC_CCD	ଆଜି RB	ଆଜି RB	ଆଜି

15	ଆସନ୍ତା JJ	ମଧ୍ୟ RP_RPD	ମଧ୍ୟ RP_RPD	ମଧ୍ୟ
16	ଟୁରିଷ୍ଟ JJ	ବଳବତ୍ତର JJ	ବଳବତ୍ତର N_NN	ବଳବତ୍ତର
17	ସିଜନରେ N_NN	ଅଛି V_VM_VF	ଅଛି V_VM_VF	ଅଛି
18	ହେଇପାରେ RB	। RD_PUNC	। RD_PUNC	।
19	ପେଲେସ୍ N_NNP			
20	ଅନ୍ N_NNP	ଆଳୁର N_NN	ଆଳୁର N_NN	ଆଳୁର
21	ହ୍ୱୀଲ୍ସରେ N_NNP	କିଛି QT_QTF	କିଛି QT_QTF	କିଛି
22	କୌଣସି DM_DMI	କିସମରେ N_NN	କିସମରେ N_NN	କିସମରେ
23	ଲିଜ N_NNP	ବହୁତ RP_INTF	ବହୁତ RP_INTF	ବହୁତ
24	ହର୍ଲେ N_NNP	ଉତ୍କଟ JJ	ଉତ୍କଟ JJ	ଉତ୍କଟ
25	କିମ୍ବା CC_CCD	ଗନ୍ଧ N_NN	ଗନ୍ଧ N_NN	ଗନ୍ଧ
26	ଅରୂଣ N_NNP	ଆସିଥାଏ V_VM_VF	ଆସିଥାଏ V_VM_VF	ଆସିଥାଏ
27	ନାୟାର N_NNP	। RD_PUNC	। RD_PUNC	।
28	ଭଳି PSP			
29	ସେଲେବ୍ରିଟି N_NN	ସମଗ୍ର JJ	ସମଗ୍ର JJ	ସମଗ୍ର
30	ନିଜର PR_PRF	ୟୁରୋପରେ N_NNP	ୟୁରୋପରେ N_NN	ୟୁରୋପରେ
31	ବିବାହ N_NN	ଆଳୁ N_NN	ଆଳୁ N_NN	ଆଳୁ
32	କରୁଥିବା V_VM_VNG	ମୁଖ୍ୟ JJ	ମୁଖ୍ୟ JJ	ମୁଖ୍ୟ
33	ଦେଖାଦେଇ V_VM	ଭୋଜନ N_NN	ଭୋଜନ N_NN	ଭୋଜନ
34	ପାରେ V_VM_VF	ଅଟେ V_VM_VF	ଅଟେ V_VM_VF	ଅଟେ
35	| RD_PUNC	। RD_PUNC	। RD_PUNC	।
36				
37	ଆଜ୍ଞା N_NN	ଭାରତର N_NNP	ଭାରତର N_NNP	ଭାରତର
38	ହଁ RP_INJ	ସିମ୍ଲାରେ N_NNP	ସିମ୍ଲାରେ N_NNP	ସିମ୍ଲାରେ
39	, RD_PUNC	କେନ୍ଦ୍ରୀୟ N_NNP	କେନ୍ଦ୍ରୀୟ JJ	କେନ୍ଦ୍ରୀୟ
40	ରେଳରେ N_NN	ଆଳୁ N_NNP	ଆଳୁ N_NN	ଆଳୁ
41	ବିବାହ N_NN	ସଂସ୍ଥାନ N_NNP	ସଂସ୍ଥାନ N_NN	ସଂସ୍ଥାନ
42	ଏବଂ CC_CCD	କୁଫେରୀ N_NNP	କୁଫେରୀ JJ	କୁଫେରୀ
43	ରୟାଲ N_NNP	ଶ୍ରେଣୀର N_NN	ଶ୍ରେଣୀର N_NN	ଶ୍ରେଣୀର
44	ରାଜସ୍ଥାନ N_NNP	ପାଖାପାଖି RB	ପାଖାପାଖି RB	ପାଖାପାଖି
45	ଅନ୍ N_NNP	୪୫ଟି RP_CL	୪୫ଟି N_NN	୪୫ଟି
46	ହ୍ୱୀଲ୍ସରେ N_NNP	କିସମ N_NN	କିସମ N_NN	କିସମ
47	ମଧୁଚନ୍ଦ୍ରିକାର N_NN	ବିକଶିତ N_NN	ବିକଶିତ N_NN	ବିକଶିତ
48	ପେକେଜ୍ N_NN	କରି V_VM_VNF	କରି V_VM_VNF	କରି
49	ଏହି DM_DMD	ଆଳୁ N_NN	ଆଳୁ N_NN	ଆଳୁ
50	ଗାଡ଼ିଗୁଡ଼ିକର N_NN	କ୍ରାନ୍ତିରେ N_NN	କ୍ରାନ୍ତିରେ N_NN	କ୍ରାନ୍ତିରେ
51	ସଞ୍ଚାଳନ N_NN	ନିଜର PR_PRF	ନିଜର PR_PRF	ନିଜର
52	ସମ୍ଭାଳୁଥିବା V_VM_VNG	ଭୂମିକା N_NN	ଭୂମିକା N_NN	ଭୂମିକା
53	ରାଜସ୍ଥାନ N_NNP	ନିର୍ବାହ N_NN	ନିର୍ବାହ N_NN	ନିର୍ବାହ
54	ପର୍ଯ୍ୟଟନ N_NNP	କରିଛନ୍ତି V_VM_VF	କରିଛନ୍ତି V_VM_VF	କରିଛନ୍ତି
55	ବିକାସ N_NNP	। RD_PUNC	। RD_PUNC	।

56	ନିଗମ N_NNP			
57	(RD_SYM	ଆଳୁର N_NN	ଆଳୁର N_NN	ଆଳୁର
58	ଆର୍‌ଟୀଡ଼ୀସୀ N_NNP	ମହତ୍ତ୍ୱକୁ N_NN	ମହତ୍ତ୍ୱକୁ N_NN	ମହତ୍ତ୍ୱକୁ
59	) RD_SYM	ଆଖୁ N_NN	ଆଖୁ N_NN	ଆଖୁ
60	ର PSP	ଆଗରେ N_NST	ଆଗରେ N_NST	ଆଗରେ
61	ଯୋଜନାଗୁଡ଼ିକରେ N_NN	ରଖି୍ V_VM_VNF	ରଖି୍ V_VM_VNF	ରଖି୍
62	ସାମିଲ N_NN	ସଂଯୁକ୍ତ N_NNP	ସଂଯୁକ୍ତ N_NNP	ସଂଯୁକ୍ତ
63	ଅଛି V_VM_VF	ରାଷ୍ଟ୍ର N_NNP	ରାଷ୍ଟ୍ର N_NNP	ରାଷ୍ଟ୍ର
64	| RD_PUNC	ସଂଘ N_NNP	ସଂଘ QT_QTC	ସଂଘ
65		ବର୍ଷ N_NN	ବର୍ଷ N_NN	ବର୍ଷ
66	ଆର୍‌ଟୀଡ଼ୀସୀ N_NNP	୨୦୦୮କୁ QT_QTC	୨୦୦୮କୁ N_NN	୨୦୦୮କୁ
67	କାୟଦା N_NN	‘ RD_PUNC	‘ RD_PUNC	‘
68	ଅନୁସାରେ PSP	ଆଳୁ N_NN	ଆଳୁ QT_QTC	ଆଳୁ
69	ନିଜର PR_PRF	ବର୍ଷ N_NN	ବର୍ଷ N_NN	ବର୍ଷ
70	ଏଜେଣ୍ଟମାନଙ୍କୁ N_NN	’ RD_PUNC	’ RD_PUNC	’
71	କହିଛନ୍ତି V_VM_VF	ଘୋଷିତ JJ	ଘୋଷିତ N_NN	ଘୋଷିତ
72	ଯେ CC_CCS	କରିଥିଲେ V_VM_VF	କରିଥିଲେ V_VM_VF	କରିଥିଲେ
73	ସେମାନେ PR_PRP	। RD_PUNC	। RD_PUNC	।
74	ହଲିଉଡ୍‌କୁ N_NNP			
75	ଏହି DM_DMD	ଦୁନିଆରେ N_NN	ଦୁନିଆରେ N_NN	ଦୁନିଆରେ
76	ପେକେଜ୍ N_NN	୧୨୫ଟି RP_CL	୧୨୫ଟି N_NN	୧୨୫ଟି
77	ପାଇଁ PSP	ଦେଶରେ N_NN	ଦେଶରେ N_NN	ଦେଶରେ
78	ଆଖିରେ N_NN	ଆଳୁର N_NN	ଆଳୁର N_NN	ଆଳୁର
79	ରଖନ୍ତୁ V_VM_VF	ଚାଷ N_NN	ଚାଷ N_NN	ଚାଷ
80	| RD_PUNC	କରାଯାଏ V_VM_VF	କରାଯାଏ V_VM_VF	କରାଯାଏ
81		। RD_PUNC	। RD_PUNC	।
82	ନୂଆଁ JJ			
83	ଯୋଜନାରେ N_NN	ଆଜି RB	ଆଜି RB	ଆଜି
84	ଆହୁରି QT_QTF	ବିଶ୍ୱରେ N_NN	ବିଶ୍ୱରେ N_NN	ବିଶ୍ୱରେ
85	ଗୋଟିଏ RP_CL	ଆଳୁର N_NN	ଆଳୁର N_NN	ଆଳୁର
86	ମୁଖ୍ୟ JJ	୫୦୦୦ QT_QTC	୫୦୦୦ N_NN	୫୦୦୦
87	କଥା N_NN	ପାଖାପାଖି RB	ପାଖାପାଖି RB	ପାଖାପାଖି
88	ଏହି DM_DMD	କିସମ N_NN	କିସମ N_NN	କିସମ
89	ରାଜକୀୟ JJ	ଅଛି V_VM_VF	ଅଛି V_VM_VF	ଅଛି
90	ଗାଡ଼ିଗୁଡ଼ିକର N_NN	ଯାହା DM_DMR	ଯାହା DM_DMR	ଯାହା
91	ଚାଲିବାର V_VM_VINF	ସର୍ବାଧିକ QT_QTF	ସର୍ବାଧିକ RP_INTF	ସର୍ବାଧିକ
92	ସମୟକୁ N_NN	ଆଣ୍ଡିଜ୍ N_NNP	ଆଣ୍ଡିଜ୍ JJ	ଆଣ୍ଡିଜ୍
93	ମଧ RP_RPD	ପାର୍ବତ୍ୟାଞ୍ଚଳରେ N_NN	ପାର୍ବତ୍ୟାଞ୍ଚଳରେ N_NN	ପାର୍ବତ୍ୟାଞ୍ଚଳରେ
94	ବଢ଼ାଇବା V_VM_VINF	ଉତ୍ପାଦିତ N_NN	ଉତ୍ପାଦିତ JJ	ଉତ୍ପାଦିତ
95	ଅଟେ V_VM_VF	ହୋଇଥାଏ V_VM_VF	ହୋଇଥାଏ V_VM_VF	ହୋଇଥାଏ

96	| RD_PUNC	। RD_PUNC	। RD_PUNC	।
97				

Appendix 4

Representative Set of Evaluation Data Format for CRF

1	**text**	**Gold**	**tagger output**
2	୧୯୭୪	QT_QTC	QT_QTC
3	ମସିହାରେ	N_NN	N_NN
4	ଏରିକଜେଙ୍କିସ୍	N_NNP	N_NN
5	ଗୋଟିଏ	RP_CL	RP_CL
6	ଗଛରୁ	N_NN	N_NN
7	୧୬୮	QT_QTC	QT_QTC
8	କିଲୋଗ୍ରାମ	N_NN	N_NN
9	ଆଳୁ	N_NN	N_NN
10	ଉତ୍ପାଦନ	N_NN	N_NN
11	କରି	V_VM_VNF	V_VM_VNF
12	ବିଶ୍ୱରେକର୍ଡ	N_NN	N_NN
13	ସୃଷ୍ଟିକରିଥିଲେ	V_VM_VF	V_VM_VF
14	ଯାହା	DM_DMR	DM_DMR
15	ଆଜି	RB	RB
16	ମଧ୍ୟ	RP_RPD	RP_RPD
17	ବଳବତ୍ତର	JJ	N_NN
18	ଅଛି	V_VM_VF	V_VM_VF
19	।	RD_PUNC	RD_PUNC
20			
21	ଆଳୁର	N_NN	N_NN
22	କିଛି	QT_QTF	QT_QTF
23	କିସମରେ	N_NN	N_NN
24	ବହୁତ	RP_INTF	RP_INTF
25	ଉତ୍କଟ	JJ	JJ
26	ଗନ୍ଧ	N_NN	N_NN
27	ଆସିଥାଏ	V_VM_VF	V_VM_VF
28	।	RD_PUNC	RD_PUNC
29			
30	ସମଗ୍ର	JJ	JJ
31	ୟୁରୋପରେ	N_NNP	N_NN
32	ଆଳୁ	N_NN	N_NN
33	ମୁଖ୍ୟ	JJ	JJ
34	ଭୋଜନ	N_NN	N_NN
35	ଅଟେ	V_VM_VF	V_VM_VF
36	।	RD_PUNC	RD_PUNC
37			
38	ଭାରତର	N_NNP	N_NNP

39	ସିମ୍ଲାରେ	N_NNP	N_NNP
40	କେନ୍ଦ୍ରୀୟ	N_NNP	JJ
41	ଆଳୁ	N_NNP	N_NN
42	ସଂସ୍ଥାନ	N_NNP	N_NN
43	କୁଫେରୀ	N_NNP	JJ
44	ଶ୍ରେଣୀର	N_NN	N_NN
45	ପାଖାପାଖି	RB	RB
46	୪୫ଟି	RP_CL	N_NN
47	କିସମ	N_NN	N_NN
48	ବିକଶିତ	N_NN	N_NN
49	କରି	V_VM_VNF	V_VM_VNF
50	ଆଳୁ	N_NN	N_NN
51	କ୍ରାନ୍ତିରେ	N_NN	N_NN
52	ନିଜର	PR_PRF	PR_PRF
53	ଭୂମିକା	N_NN	N_NN
54	ନିର୍ବାହ	N_NN	N_NN
55	କରିଛନ୍ତି	V_VM_VF	V_VM_VF
56	।	RD_PUNC	RD_PUNC
57			
58	ଆଳୁର	N_NN	N_NN
59	ମହତ୍ତ୍ୱକୁ	N_NN	N_NN
60	ଆଖୁ	N_NN	N_NN
61	ଆଗରେ	N_NST	N_NST
62	ରଖି	V_VM_VNF	V_VM_VNF
63	ସଂଯୁକ୍ତ	N_NNP	N_NNP
64	ରାଷ୍ଟ୍ର	N_NNP	N_NNP
65	ସଂଘ	N_NNP	QT_QTC
66	ବର୍ଷ	N_NN	N_NN
67	୨୦୦୮କୁ	QT_QTC	N_NN
68	‘	RD_PUNC	RD_PUNC
69	ଆଳୁ	N_NN	QT_QTC
70	ବର୍ଷ	N_NN	N_NN
71	’	RD_PUNC	RD_PUNC
72	ଘୋଷିତ	JJ	N_NN
73	କରିଥିଲେ	V_VM_VF	V_VM_VF
74	।	RD_PUNC	RD_PUNC
75			
76	ଦୁନିଆରେ	N_NN	N_NN
77	୧୨୫ଟି	RP_CL	N_NN
78	ଦେଶରେ	N_NN	N_NN
79	ଆଳୁର	N_NN	N_NN
80	ଚାଷ	N_NN	N_NN
81	କରାଯାଏ	V_VM_VF	V_VM_VF
82	।	RD_PUNC	RD_PUNC

83			
84	ଆଜି	RB	RB
85	ବିଶ୍ୱରେ	N_NN	N_NN
86	ଆଳୁର	N_NN	N_NN
87	୫୦୦୦	QT_QTC	N_NN
88	ପାଖାପାଖି	RB	RB
89	କିସମ	N_NN	N_NN
90	ଅଛି	V_VM_VF	V_VM_VF
91	ଯାହା	DM_DMR	DM_DMR
92	ସର୍ବାଧିକ	QT_QTF	RP_INTF
93	ଆଣ୍ଡିଜ୍	N_NNP	JJ
94	ପାର୍ବତ୍ୟାଞ୍ଚଳରେ	N_NN	N_NN
95	ଉତ୍ପାଦିତ	N_NN	JJ
96	ହୋଇଥାଏ	V_VM_VF	V_VM_VF
97	।	RD_PUNC	RD_PUNC
98			
99	ଗୋଟିଏ	RP_CL	RP_CL
100	ଭଲ	JJ	JJ
101	କଥା	N_NN	N_NN
102	ଏହା	DM_DMD	DM_DMD
103	ଅଟେ	V_VM_VF	V_VM_VF
104	ଯେ	CC_CCS	CC_CCS
105	ଅନ୍ତର୍ରାଷ୍ଟ୍ରୀୟ	JJ	JJ
106	ଆଳୁ	N_NN	N_NN
107	ଅନୁସନ୍ଧାନ	N_NN	N_NN
108	କେନ୍ଦ୍ରର	N_NN	N_NN
109	ଜିନ୍	N_NN	N_NN
110	ବ୍ୟାଙ୍କରେ	N_NN	N_NN
111	ଏହି	DM_DMD	DM_DMD
112	ସବୁ	DM_DMI	DM_DMI
113	କିସମ	N_NN	N_NN
114	ସୁରକ୍ଷିତ	JJ	JJ
115	ଅଛି	V_VM_VF	V_VM_VF
116	।	RD_PUNC	RD_PUNC

BIBLIOGRAPHY

Books and Encyclopedias:

Abbi, A. (2001). *A Manual of Linguistic Fieldwork and Structures of Indian languages* (Vol. 17). Lincom Europa.

Bharati, A., Chaitanya, V., & Sangal, R. (2004). *Natural language Processing: A Paninian Perspective,* Prentice Hall of India Private Limited, New Delhi.

Brown, K. (2006). Encyclopedia of Language and Linguistics. National Research Council of Canada: Elsevier.

Crystal, D. (1992). *An Encyclopedic Dictionary of Language and Languages*. U. K.: Blackwell Publishers, Oxford.

Jain, D. & Cardona, G. (2007). The Indo-Aryan Languages. Taylor & Francis.

Jurafsky, D., & Martin, J. H. (2002). *Speech and Language Processin*g, Pearson Education, Delhi, p.318

Kidwai, A. (2000). *XP-adjunction in Universal Grammar*. New York: Oxford University Press.

Kachru, Y. (2006). Hindi (Vol. 12). Amsterdam: John Benjamin’s Publishing.

Koul, O. N. (2008). Modern Hindi Grammar. USA: Dunwoody Press.

Mahapatra, D. (1997). *Oriya Dhwanitatwa o Sabda Sambhar*, Cuttack: Friends’ Publication.

Neukom, L., & Patnaik, M. (2003). A grammar of Oriya. Seminar für Allgemeine Sprachwissenschaft der Univ. Zürich.

Majumdar, P.C. (1970). *A Historical Phonology of Odia*. Calcutta: Sanskrit College.

Masica, C.P. (1991). The Indo-Aryan Languages. Cambridge: CUP.

Mitkov, R. (2003). *The Oxford Handbook of Computational Linguistics.* Oxford University Press, New York.

Sober, M. M., & Benedito, J. R. M. (Eds.). (2010). *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*. Information Science Reference.

Subbarao, K.V. (2012). *South Asian Languages: A Syntactic Typology*. Cambridge University Press.

Sutton, A. (1831). *An Introductory Grammar of the Oriya Language*. Printed at the Baptist Mission Press.

Tripathi, K. B. (1962). *The Evolution of Odia Language and Script.* Cuttack: Utkal University.

Theses and dissertation:

Banerjee, E. (2011). *Intra-phrasal Agreement within the Hindi Noun Phrase: A Grammar Checker Approach*. M.Phil. Dissertation, Centre for Linguistics, J.N.U., New Delhi.

Chandrashekar, R. (2007). *Part-of-Speech Tagging for Sanskrit***,** Ph.D. Thesis, Special Centre for Sanskrit Studies, J.N.U., New Delhi.

Choudhary, N. (2006). *Developing a Computational Framework for the Verb Morphology of Great Andamanese*. M.Phil. Dissertation, Centre for Linguistics, J.N.U., New Delhi.

Hardie, A. (2003). *The Computational Analysis of Morpho-syntactic Categories in Urdu*. Ph.D. thesis submitted to the Dept. of Linguistics and Modern English, Lancaster University, (revised soft copy 2004), p. 40.

Humayoun, M. (2006). *Urdu Morphology, Orthography and Lexicon Extraction*. Master’s Thesis. Department of Computing Science, Chalmers University of Technology.

Majhi, T. D. (2007). *Descriptive Oriya Morphology in the Paninian Model.* Ph.D. Thesis, Centre for Linguistics, J.N.U., New Delhi.

Megyesi, B. (1998). *Brill’s rule-based part of speech tagger for Hungarian. Master's thesis, University of Stockholm.*

Singh, D. P. (2011). *A Comparative Study of Hindi Parts of Speech Tagsets*. M.Phil Dissertation, Centre for Linguistics, J.N.U., New Delhi.

Sahoo, K. (1996). *The DP Analysis in Oriya*. M.Phil. Dissertation, CIEFL, Hyderabad.

Uniyal, A. (2011). *Issues and Challenges in Hindi Shallow Parsing*. M.Phil. Dissertation, Centre for Linguistics, J.N.U., New Delhi.

DeRose, S. J. (1990). *Stochastic Methods for Resolution of Grammatical Category Ambiguity in Inflected and Unified Languages*. Ph.D. Dissertation. Department of Cognitive and Linguistic Sciences, Providence, Brown University.

Reports:

Pattanayak, D. P. and Prushty, S. K. (2013). Classical Odia Language, KIS Foundation, Bhubaneswar. Retrieved on 06.06.15 from http://www.orissalinks.com/odia/classical1.pdf

State of the Environment Report- Orissa, 2007. Retrieved on 09.06.15 from http://envfor.nic.in/soer/state/SoE-orissa.pdf

Papers:

Abbi, A. (1991). *Semantics of explicator compound verbs*. In South Asian Languages, Language Sciences, 13:2, 161-180.

Ali, H. (2010). An unsupervised parts-of-speech tagger for the Bangla language. *Department of Computer Science, University of British Columbia*.

Antony, P. J., & Soman, K. P. (2012). Computational morphology and natural language parsing for Indian languages: a literature survey. *International Journal of Computer Science and Engineering Technology (IJCSET), 3*, 136-146.

___________, Antony, P. J., Mohan, S. P., & Soman, K. P. (2010, March). SVM Based Part of Speech Tagger for Malayalam. In *Recent Trends in Information, Telecommunication and Computing (ITC), 2010 International Conference on* (pp. 339-341). IEEE.

___________, & Soman, K. P. (2011). Parts of speech tagging for Indian languages: a literature survey. *International Journal of Computer Applications (0975-8887), 34*(8).

Banerjee, E., Kaushik, S., Nainwani, P., Bansal, A. and Jha, G. N. (2013). Linking and Referencing Multi-lingual corpora in Indian languages, Poland: 6th LTC Conference, 65-68.

Baskaran, S., Bali, K., Bhattacharya, T., Bhattacharyya, P., & Jha, G. N. (2008). A common parts-of-speech tagset framework for indian languages. In *In Proc. of LREC 2008*.

Brants, T. (2000). TnT: a statistical part-of-speech tagger. In *Proceedings of the sixth conference on applied natural language processing* (pp. 224-231). Association for Computational Linguistics.

Bright, W. (1996). The Devanagari script. In Peter Daniels and William Bright, editors, The World's Writing Systems. Oxford University Press, New York, NY, pages 384–390.

Brill, E. (1992, February). A simple rule-based part of speech tagger. In *Proceedings of the workshop on Speech and Natural Language* (pp. 112-116). Association for Computational Linguistics.

Chakrabarti, D., & CDAC, P. (2011). Layered parts of speech tagging for Bangla. *Language in India, Special Volume: Problems of Parsing in Indian Languages*. *www.languageinindia.com*

Chandra, N., Kumawat, S. and Srivastava, V. (2014, March). Various tagsets for Indian languages and their performance in part of speech tagging. In *Proceedings of 5th IRF International Conference,* Chennai.

Choudhary, N., & Jha, G. N. (2014). Creating Multilingual Parallel Corpora in Indian Languages. In *Human Language Technology Challenges for Computer Science and Linguistics* (pp. 527-537). Springer International Publishing.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning, 20*(3), 273-297.

Dandapat, S., Sarkar, S., & Basu, A. (2004, December). A Hybrid Model for Part-of-Speech Tagging and its Application to Bengali. In *International conference on computational intelligence* (pp. 169-172).

Das, B. R., & Patnaik, S. (2014, January). 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 International Publishing. Retrieved on 09.06.15 from http://soauniversity.ac.in/home/research/publications/novel-approach-odia-part-speech-tagging-using-artificial-neural-network-1

__________, Sahoo, S., Panda, C. S., & Patnaik, S. (2015). Part of Speech Tagging in Odia Using Support Vector Machine. *Procedia Computer Science, 48*, 507-512.

Deb, Debjit. (2012). On Case Marking in Assamese, Bengali, and Oriaya. IJAL &EL.

Ekbal, A., Mondal, S., & Bandyopadhyay, S. (2007). POS Tagging using HMM and Rule-based Chunking. *The Proceedings of SPSAL*, 25-28.

__________, & Bandyopadhyay, S. (2008, December). Part of speech tagging in Bengali using support vector machine. In *Information Technology, 2008. ICIT'08. International Conference on* (pp. 106-111). IEEE.

__________, Haque, R., & Bandyopadhyay, S. (2007, December). Bengali part of speech tagging using conditional random field. In *Proceedings of Seventh International Symposium on Natural Language Processing (SNLP2007)* (pp. 131-136).

Gill, M. S., Lehal, G. S., & Joshi, S. S. (2009). Part of speech tagging for grammar checking of Punjabi. *The Linguistic Journal*, *4*(1), 6-21.

Hardie, A. (2005). Automated part-of-speech analysis of Urdu: conceptual and technical issues.

__________. (2003). Developing a tagset for automated part-of-speech tagging in Urdu. Department of Linguistics, Lancaster University. Retrieved on 02.06.2014 from www.lancs.ac.uk/staff/hardiea/c103 urdu.pdf

He, X., Zemel, R. S., & Carreira-Perpindin, M. A. (2004). Multiscale conditional random fields for image labelling. In *Computer vision and pattern recognition, 2004. CVPR, 2004. Proceedings of the 2004 IEEE computer society conference on* (Vol. 2, pp. II-695). IEEE.

Hellwig, O. (2009). Sanskrit tagger: A stochastic lexical and POS tagger for Sanskrit. In *Sanskrit Computational Linguistics* (pp. 266-277). Springer Berlin Heidelberg.

Jena, I., Chaudhury, S., Chaudhry, H., & Sharma, D. M. (2011). Developing Oriya Morphological Analyzer Using Lt-toolbox. In *Information Systems for Indian Languages* (pp. 124-129). Springer Berlin Heidelberg.

Jha, G. N. (2010). The TDIL program and the Indian Language Corpora Initiative (ILCI). In *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC 2010). European Language Resources Association (ELRA).*

__________, Beermann, D., Hellan, L., Singh, S., Behera, P. and Banerjee, E. (2014). Indian languages on the TypeCraft platform– the Case of Hindi and Odia. In *the proceedings of the LREC-2014*, Rekyavik.

Joachims, T. (1999). Making large scale SVM learning practical. In: Schölkopf B, Burges C, Smola A (eds.) Advances in kernel methods- support vector learning. MIT Press, Cambridge.

Klinger, R., & Tomanek, K. (2007). *Classical probabilistic models and conditional random fields*. TU, Algorithm Engineering.

Kumar, D., & Josan, G. S. (2010). Part of speech taggers for morphologically rich Indian languages: a survey. *International Journal of Computer Applications (0975–8887) Volume*, 1-9.

Kumar, R., Kaushik, S., Nainwani, P., Banerjee, E., Hadke, S., & JHA, G. N. (2012, March). Using the ILCI annotation tool for POS annotation: A case of Hindi. In *13th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2012), New Delhi, India*.

Lafferty, J., McCallum, A., & Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of the 18th International Conference on Machine Learning*. Morgan Kaufmann. pp. 282–289.

Leech, G., & Wilson, A. (1996). EAGLES: Recommendations for the Morphosyntactic Annotation of Corpora (EAGLES Document EAG–TCWG–MAC/R). *Pisa, Consiglio Nazionale delle Ricerche. Istituto di Linguistica Computazionale*.

Mahapatra, B. P. (1996). Oriya writing. In Peter Daniels and William Bright, editors, *The World's Writing Systems*. Oxford University Press, New York, NY, pages 404–407.

Manning, C. D. (2011). Part-of-speech tagging from 97% to 100%: is it time for some linguistics?. In *Computational Linguistics and Intelligent Text Processing* (pp. 171-189). Springer Berlin Heidelberg.

Mishra, N., & Mishra, A. (2011, June). Part of Speech Tagging for Hindi Corpus. In *Communication Systems and Network Technologies (CSNT), 2011 International Conference on* (pp. 554-558). IEEE.

Muaz, A., Ali, A., & Hussain, S. (2009, August). Analysis and development of Urdu POS tagged corpus. In *Proceedings of the 7th Workshop on Asian Language Resources* (pp. 24-29). Association for Computational Linguistics.

Nainwani, P., Banerjee, E., Kaushik, S., & Jha, G. N. (2012). Issues in annotating less resourced languages- the case of Hindi from Indian Languages Corpora Initiative (ILCI).

Pathak, P., Patel, P., Panchal, V., Choudhary, N., Patel, A., & Joshi, G. (2014). ezDI: A Hybrid CRF and SVM based Model for Detecting and Encoding Disorder Mentions in Clinical Notes. *SemEval 2014*, 278.

Patnaik, B. N. (2001). Nominative and non-nominative constructions in Oriya. Retrieved on 07.02.2010 from http://home.iitk.ac.in/~patnaik/documents/nnom.pdf

Patra, B. G., Kumbhar, D., Das, D., Bandyopadhyay, S. (2012, December). Part of Speech (POS) Tagger for Kokborok. In *24th International Conference on Computational Linguistics*, 923.

Rajendran, K., Saravan, L. Sobha, and Subbarao, K. V. S. (2008). A Common Parts of Speech Tagset Framework for Indian Languages. In *Proceedings of the 6th Language Resources and Evaluation Conference (LREC)*.

RamaSree, R. J., & Kusuma K. P. (2007). Combining pos taggers for improved accuracy to create Telugu annotated texts for information retrieval. *Department of Telugu Studies, Tirupathi, India*.

Rao, P. R., Vijay, S. R., Vijaya Krishna, R., & Shoba, L. (2007). A text chunker and hybrid POS tagger for Indian Languages. In *the Proceedings of IJCAI workshop on "Shallow parsing for south Asian languages*.

Sahoo, A. (2010). Oriya passives with di-transitives. Papers from the Lancaster University Postgraduate Conference in Linguistics & Language Teaching.

Sahoo, K. (2003, October). Oriya nominal forms: a finite state processing. In *TENCON 2003. Conference on Convergent Technologies for the Asia-Pacific Region* (Vol. 2, pp. 730-734). IEEE.

Sajjad, H., & Schmid, H. (2009, March). Tagging Urdu text with parts of speech: A tagger comparison. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 692-700). Association for Computational Linguistics.

Santorini, B. (1990). Parts-of-Speech Tagging Guidelines for the Penn Treebank Project. Third Revision, Second Printing.

Schiilkop, P. B., Burgest, C., & Vapnik, V. (1995). Extracting support data for a given task. *no. x*.

Settles, B. (2004, August). Biomedical named entity recognition using conditional random fields and rich feature sets. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications* (pp. 104-107). Association for Computational Linguistics.

Sethi, D. P. (2013 August). Baleswari Odia dialect identification using Rule Based technique. Vol. 2, Issue. 8, (pp. 466-471). *International Journal of Computational Linguistics and Natural Language Processing (IJCLNLP).*

__________. (2013 October). Morphological analyzer for Sambalpuri Odia dialect inflected verbal forms. Vol. 3, Issue 10. *International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE).*

__________. (2014 March). A survey on Odia computational morphology. Vol. 3, Issue. 3, (pp. 623-625). *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET).*

Sha, F., & Pereira, F. (2003, May). Shallow parsing with conditional random fields. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1* (pp. 134-141). Association for Computational Linguistics.

Sharma, S. K., & Lehal, G. S. (2011, June). Using Hidden Markov Model to improve the accuracy of Punjabi POS tagger. In *Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on* (Vol. 2, pp. 697-701). IEEE.

Shrivastava, M., & Bhattacharyya, P. (2008, December). Hindi POS tagger using naive stemming: harnessing morphological information without extensive linguistic knowledge. In *International Conference on NLP (ICON08), Pune, India*.

Toutanova, K., Klein, D., Manning, C. D., & Singer, Y. (2003, May). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1* (pp. 173-180). Association for Computational Linguistics.

Vapnik, V. N. & Vapnik, V. (1998). *Statistical learning theory* (Vol. 1). New York: Wiley.

Vaz, E., Walawalikar, S. V., Pawar, J., and Sardesai, M. (2012, December). BIS Annotation Standards with Reference to Konkani Language. In *Proceedings of the*

3rd Workshop on South and Southeast Asian Natural Language Processing (SANLP), pp. 145-152, COLING. Mumbai.

Xiao, R. Z., McEnery, A. M., Baker, J. P., & Hardie, A. (2004, March). Developing Asian language corpora: standards and practice. In *The 4th Workshop on Asian Language Resources*.

Internet Sources:

http://www.censusindia.gov.in/2011documents/lsi/ling_Orissa.html

www.ethnologue.com

https://catalog.ldc.upenn.edu/LDC2000T43

http://corpus.byu.edu/coca/

http://www.natcorp.ox.ac.uk/

http://corpus.byu.edu/time/

http://ildc.in/Oriya/Oindex.aspx

http://ltrc.iiit.ac.in/showfile.php?filename=ltrc/internal/nlp/corpus/index.html

http://www.cdac.in/index.aspx?id=mc_ilf_indian_language_fcdt

http://www.orissatourism.org/orissa-geography.html

http://oriya.indiatyping.com/index.php/download-oriya-font

http://www.odialanguage.com/Odia_fonts.html

http://www.aparts.org/products/aprant-font-odia-key-board-manager/odia-fonts/

http://www.aparts.org/products/aprant-font-odia-key-board-manager/odia-fonts/

http://sanskrit.jnu.ac.in/projects/ilci.jsp?proj=ilci

http://www.ciil.org/ProgReportworkshop3.aspx

http://www.tdil-dc.in/

http://www.ldcil.org/resourcesSpeechCorpOriya.aspx

http://www.ldcil.org/resourcesSpeechCorpOriya.aspx http://www.ldcil.org/resourcesTextCorp.aspx

http://www.imagact.it/imagact/query/dictionary.seam

http://typecraft.org/tc2wiki/Main_Page

http://indradhanush.unigoa.ac.in/odiawordnet

http://www.comp.lancs.ac.uk/ucrel/claws/trial.html

http://www.coli.uni-saarland.de/~thorsten/tnt/

http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

http://nlp.stanford.edu/software/tagger.shtml

http://wortschatz.uni-leipzig.de/~cbiemann/software/unsupos.html.

http://ilk.uvt.nl/mbt/

http://ucrel.lancs.ac.uk/claws/

http://sanskrit.jnu.ac.in/rstudents/chandra_thesis.pdf

http://sanskrit.jnu.ac.in/post/post.jsp

http://www.indsenz.com/int/index.php?content=sanskrit_tagger

http://www.ling.gu.se/~lager/mogul/brill-tagger/

https://catalog.ldc.upenn.edu/LDC2000T43

http://corpus.byu.edu/coca/

http://www.natcorp.ox.ac.uk/

http://corpus.byu.edu/time/

http://ildc.in/Oriya/Oindex.aspx)

http://www.oracle.com/technetwork/java/javaee/servlet/index.html

http://www.serverwatch.com/news/article.php/1125001/Apache-Tomcat-40-Final-Released.htm

http://www.oracle.com/technetwork/java/javaee/servlet/index.html http://www01.sil.org/linguistics/glossaryoflinguisticterms/WhatIsAnAgglutinativeLanguage.html