

DESIGN AND DEVELOPMENT OF DEVANAGARI SCRIPT HANDWRITTEN TEXT CORPUS WITH ANNOTATION

Ph.D. Thesis

MANINDER SINGH NEHRA

(ID No. 2013RCP9553)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MALAVIYA NATIONAL INSTITUTE OF TECHNOLOGY JAIPUR

December, 2019

Design and Development of Devanagari Script Handwritten Text Corpus with Annotation

submitted in

fulfillment of the requirements for the degree of

Doctor of Philosophy

by

Maninder Singh Nehra

ID: 2013RCP9553

Under the Supervision of

Dr. Neeta Nain

Dr. Mushtaq Ahmed



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MALAVIYA NATIONAL INSTITUTE OF TECHNOLOGY JAIPUR, INDIA

December, 2019

CERTIFICATE

This is to certify that the thesis entitled, “**DESIGN AND DEVELOPMENT OF DEVANAGARI SCRIPT HANDWRITTEN TEXT CORPUS WITH ANNOTATION**” being submitted by **Maninder Singh Nehra (2013RCP9553)** is a bonafide research work carried out under our supervision and guidance in fulfillment of the requirement for the award of the degree of **Doctor of Philosophy** in the Department of Computer Science & Engineering, Malaviya National Institute of Technology Jaipur, India. The matter embodied in this thesis is original and has not been submitted to any other university or institute for the award of any other degree.

Place: Jaipur

Date:

Dr. Neeta Nain

(Supervisor)

Associate Professor

Computer Science and Engineering

MNIT Jaipur

Dr. Mushtaq Ahmed

(Co-Supervisor)

Associate Professor

Computer Science and Engineering

MNIT Jaipur

DECLARATION

I, **Maninder Singh Nehra**, declare that this thesis titled, “**DESIGN AND DEVELOPMENT OF DEVANAGARI SCRIPT HANDWRITTEN TEXT CORPUS WITH ANNOTATION**” and the work presented in it, are my own, I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this university.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this university or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always give. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself, jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date:

Maninder Singh Nehra
(2013RCP9553)

ACKNOWLEDGMENTS

It gives me immense pleasure to express gratitude and regards to all those people who supported me for this doctoral research work at MNIT Jaipur. I would first like to thank God who gave me the grace and privilege to pursue this program. I express my heart-felt gratefulness to my supervisors Dr. Neeta Nain and Dr. Mushtaq Ahmed for perpetual inspiration, valuable advices, enormous support and blessings. It is needless to say without them, this research work would have not been possible. The best time and fruitful discussion with them have immense contributions in the process of completion of my Ph.D work that would be treasured throughout my life.

My special thanks to Departmental Research Committee (DREC) members, Dr. Neeta Nain, Dr. Mushtaq Ahmed, Dr. Girdhari Singh and Dr. Namita Mital for their valuable criticisms and worthwhile suggestions. I am also grateful to Dr. Yogesh Meena, Dr. Dinesh Gopalani, all other faculty members of the department for their inspiration and suggestions at various stages. Let me express special thanks to Head of the Department, Dr. Girdhari Singh and Dr. M.C. Govil. I am extremely thankful to Prof. Udaykumar Yaragatti, Director, MNIT Jaipur for providing me infrastructural facilities to work. I would also like to thank all the referees who reviewed this work as pieces of it were submitted to international journals and conferences.

I would like to thank my co-scholars Prakash, Tapas, Ravindra, Abhishek, Shweta, Sonu, Deepa and other research colleagues of the department for their loving cooperation and to spend quality time together. Let me also mention colleague at Govt. Engineering College Bikaner, Dr. J.P. Bhamu, Dr. Sanjeev Jain, Dr. Mahendra Bhadu for their inspiration and suggestions at various stages. I feel a deep sense of gratitude for my Late parents, mother, sisters and my brothers Sanjay Singh and Banwari Lal. My lovely, daughter-Deveshi and son-Dhanish and my caring wife Manju.

Finally, I gratefully, acknowledge one and all who are directly or indirectly involved to shape this research work.

Contents

Abstract	i
List of Figures	iii
List of Tables	v
1 Introduction	1
1.1 Linguistics of Corpus	1
1.1.1 Annotation of Corpus	2
1.1.2 Statistical Profiling	3
1.2 Motivation	4
1.3 Challenges	8
1.4 Aims and Objectives	8
1.5 Thesis Organization	10
2 Literature Survey of Corpus Linguistic	12
2.1 Introduction	12
2.2 Corpus Linguistics Perspective	13
2.2.1 Corpus Linguistic Evolution	13
2.2.2 Corpus Linguistic in Linguists View	14
2.3 Principle of Corpus Linguistic	15
2.3.1 Salient Features of Corpus	15
2.3.2 Taxonomy of Corpus	17
2.3.2.1 Genre of Text	17
2.3.2.2 Nature of Text	19
2.3.2.3 Type of Text	20
2.3.3 Representativeness	20
2.3.4 Purpose of Design	21
2.4 Survey of Corpora	22
2.4.1 Handwritten Corpus of English, Spanish, Indonesian, Ko- rean and Chinese Languages	23
2.4.2 Handwritten Corpus of Chinese, Tibetan, Arabic-Latin, Japanese, Greek, Urdu and Farsi Languages	25
2.4.3 Handwritten Corpus of Arabic, Urdu, English, Japanese and Chinese Languages	26
2.4.4 Indic Corpus Augmentation	27

2.4.5	Indic Handwritten Corpus	29
2.4.6	Devanagari Handwritten Corpus	30
2.5	Summary	31
3	Corpus Compilation and Annotation Tools	33
3.1	Introduction	33
3.2	Linguistic Annotation	34
3.2.1	Annotation of Corpus	34
3.2.2	Taxonomy of Annotation	36
3.2.3	Hierarchy of Linguistic Annotation	39
3.3	Corpus Annotation Standards	40
3.3.1	Annotations should be Separable	40
3.3.2	Explicit Documentation	40
3.3.3	Encoding of Annotation	41
3.3.3.1	Text Encoding Initiative	42
3.3.3.2	Corpus Encoding Standard with XML	42
3.3.3.3	ISO Technical Committee-37/ Sub-committee-4	42
3.4	Corpus Development and Analysis Models	43
3.4.1	Automatic and Manual Analysis	43
3.4.1.1	Prevail for Annotation	44
3.4.1.2	Prevail for Annotation and Information Extraction	45
3.4.1.3	Prevail for Annotation and Statistical Analysis of Data	47
3.5	Summary	48
4	Compilation of Devanagari Script Corpus	49
4.1	Introduction	49
4.2	History of Devanagari Hindi	51
4.3	Motivation Behind Selection of Devanagari Script in Hindi	52
4.4	Features of Devanagari Script in Hindi	54
4.5	Collection and Distribution of Data	56
4.5.1	Genre of Texts	56
4.5.2	Design of Handwritten Form	57
4.6	Statistics of the Database	59
4.6.1	Category Wise Distribution of Text Page	60
4.6.2	Geographical Region Wise Selection of Writers	60
4.6.3	Text Distribution Among Category	61
4.6.4	Demographic Statistics	62
4.7	Summary	63
5	Corpus Framework for Encoding and Annotation	64
5.1	Introduction	64
5.2	Database System	65
5.2.1	Characteristics and Benefits	65
5.2.2	The Relational Model of Database	65

5.2.3	Our Corpus Database	66
5.2.4	Entity-Relationship Model (ER Model)	67
5.3	Our Model for Structural Mark-up and Linguistic Annotation	68
5.3.1	Structural Mark-up and Corpus Annotation	70
5.3.2	Structural Annotation and Auto-indexing	70
5.3.3	Validation and Ground-truthing	71
5.3.4	Coordinates Information based Mapping	74
5.3.4.1	Mark-up of Text-form	75
5.3.4.2	Mark-up of Line	76
5.3.4.3	Mark-up of Word	76
5.4	Corpus Encoding Standards	77
5.4.1	Representation of Extensible Mark-up Language (XML)	77
5.4.2	XML and Corpus Encoding	78
5.4.3	Procedure of the Corpus Encoding	79
5.5	The User Interface	80
5.5.1	Functions of DSHTC	81
5.5.2	Navigation and Information Extraction	83
5.5.3	Corpus Based Education	84
5.5.3.1	Corpus Based Dictionary	84
5.5.3.2	Corpus Based E-Learning	84
5.6	Comparative Analysis of the Our Corpus Framework	85
5.6.1	Ground-Truth Comparative Analysis	85
5.6.2	Functionality Based Comparative Analysis	88
5.7	Summary	91
6	Part-of-Speech Tagging	93
6.1	Introduction	93
6.2	Existing Work	94
6.2.1	POS Tagger for Indian Languages	95
6.2.2	POS Tagger for Hindi Language	97
6.2.3	POS Tagger for Other Languages	100
6.3	Techniques for POS Tagging	102
6.3.1	Rule Based Approach	102
6.3.2	Stochastic Approach	103
6.3.2.1	Unigram Model	104
6.3.2.2	Hidden Markov Model	105
6.3.2.3	Maximum Entropy Markov Model	106
6.3.2.4	Conditional Random Fields	107
6.3.2.5	Memory Based Model	108
6.3.3	Hybrid Approach	108
6.4	Our Part-of-Speech Tagger	108
6.4.1	System Functionalities	110
6.4.2	Unigram Model	110
6.4.3	Rules Based Model	111

6.4.3.1	Rules Based Upon Lexical Features	111
6.4.3.2	Rules Based Upon Contextual Features	114
6.4.4	Ambiguity Removing	115
6.4.5	Experimental Setup of POS Tagging Method	115
6.4.6	Performance Result of POS Tagging Method	118
6.5	Summary	120
7	Chunking	123
7.1	Introduction	123
7.2	Existing Work	125
7.2.1	Chunking for Indian Languages	126
7.2.2	Chunking for Hindi Language	128
7.2.3	Chunker for Other Languages	130
7.3	Techniques for Chunking	131
7.4	Our Method for Chunking	132
7.4.1	Categories of Chunking	133
7.4.2	Generation of Rules	134
7.4.2.1	Rules Generation for Noun Groups	135
7.4.2.2	Rules Generation for Verbs Groups	136
7.4.2.3	Rules Generation for Other Groups	136
7.4.2.4	Some Special Cases	137
7.4.3	Experimental Setup of The Chunker	137
7.4.3.1	Chunk Boundary Detector	137
7.4.3.2	Labeling of Chunk	138
7.4.4	Results and Performance Analysis	139
7.5	Summary	143
8	Statistical Analysis of Corpus	144
8.1	Introduction	144
8.2	Study of Corpus Statistics	145
8.3	Statistical Measurement of Corpus	145
8.3.1	Frequency Profiling	146
8.3.2	Entropy and Perplexity Measurement	147
8.3.3	Degree of Dispersion Measurement	149
8.3.3.1	Dispersion Measure using the Frequency Profiling	150
8.3.3.2	Dispersion Measure using the Standard Deviation	152
8.4	Word Frequency Distribution using Zipfs Rule	155
8.5	Text Line Segmentation Results	156
8.6	Summary	157
9	Conclusions and Future Work	160
9.1	Thesis Summary	160
9.2	Major Contributions of Research	162
9.3	Limitations and Future Scope	165

Appendix	165
A List of Publications	166
Bibliography	168

ABSTRACT

The corpora is a worthwhile resource for linguistic research, increasing demand for the linguistic resources tends the linguists towards the corpus. An ample number of corpus exist for the English, Chinese, Arabic and European languages. The very poor availability of the Devanagari Hindi Corpora, with only a few handwritten data sets, lead to the need for more significant research on the Devanagari script Hindi.

Devanagari script Hindi is the official language of India and fourth most frequently used language in the world. However, Hindi has complex writing nature due to its modifiers and very poor resources. Therefore, it remains a thrust research domain in document image analysis. In the view of thirstily need, a unified approach and model, Devanagari Script Handwritten Text Corpus with Annotation (DSHTC) for advancing the compilation and statistical analysis of Devanagari script Hindi is highlighted.

Our corpus consist of 1,650 handwritten text forms, 5,043 number of lines and 1,23,750 number of words, collected with 7 categories and 19 subcategories. Data collection forms were filled by 1,650 different writers of different age, gender, with diverse educational backgrounds and from different geographical regions. The compilation of corpus contains various amount of significant information such as printed texts, handwritten texts, digits, etc. which make it useful to fulfil the different requirements of Natural Language Processing (NLP) research domain.

For the systematic integration and indexing of the corpus information a framework DSHTC has been designed to annotate handwritten Devanagari script Hindi document. That includes some level of automation with aligning transcription, structural (ground-truth) and linguistic (Part-of-Speech (POS) Tagging and Chunking) annotation. POS tagging method is designed with the probabilistic approach and a rule-based approach. An ambiguity removing module is also provided in the approach. Chunking approach for Devanagari Hindi text uses the POS information of the text as contextual information. The approach is based on rule-based model. These rules are generated by detailed analysis of POS tagged corpus of Devanagari Hindi language and represented through regular expressions. Various FSM automatons are used to implement these rules.

Our framework generates an auto-indexing number at each level (text-page, line and word) and conversely indexed both Unicode and textual contour pixels in-

formation into a database with auto-generated Unique Identifier (UID), with an XML representation.

This corpus would be very beneficial for linguistic research in benchmarking and providing a testbed for handwritten text recognition techniques evaluation for Devanagari Hindi, digital forensics writer identification, signature verification, classification of handwritten and printed text, categorization of texts etc.

In addition to compilation, annotation and benchmarking test, framework engenders the word frequency distribution according to category sapient that is utilizable for linguistic evaluation. This study presents the methodology used for linguistic statistics analysis with corpus data set. The observation of statistical analysis is conducted utilizing fundamental statistics likes rank of words, frequency of words, and entropy analysis of corpus. Besides the traditional corpus statistics coverage, some additional adaptive features were withal presaged utilizing Zipf's linguistic law and measurement of dispersion in corpus information. To strengthen the claim and applicability of our framework, the experimental results obtained from statistical analysis observation on our handwritten corpus are presented to meet the requirement of linguistics.

List of Figures

2.1	Conceptual classification of the corpora.	17
2.2	Text based classification of the corpora.	18
3.1	Process for generating an annotation model.	35
4.1	The India map show major three regions according to majority of the languages.	50
4.2	A map showing the regions of the Hindi language.	52
4.3	Census data of India motivated us to work on Devanagari Hindi from Patrika news paper.	53
4.4	A set of Hindi handwritten consonants for Devanagari script in Hindi.	54
4.5	A sample of Devanagari Hindi handwritten vowels and modifiers (Matras).	55
4.6	Devanagari Hindi handwritten digits.	55
4.7	A sample filled in form.	58
4.8	A sample blank in form.	59
4.9	Domain wise distribution of handwritten data collection forms.	60
4.10	Geographical regions wise selection of writers.	61
4.11	Domain wise data distribution of lines and words.	62
5.1	ER model of the corpus database.	67
5.2	The process for corpus building, compilation, structural & linguistic annotation and statistical analysis.	69
5.3	A process of the corpus development and its annotation.	71
5.4	Automatically UID genesis of a handwritten text document.	72
5.5	The hierarchical iterative process of different levels XML ground-truth generation to annotate the handwritten text-form.	73
5.6	A sample of handwritten text form structural mark-up.	73
5.7	Screen shot of form level validation.	74
5.8	Screen shot of line level validation.	74
5.9	Screen shot to check the null value validation.	75
5.10	Model-based query operations block diagram.	81
5.11	A graphical user interface of Devanagari script handwritten text corpus.	82
6.1	Corpus annotation levels.	95

6.2	Various techniques for POS tagging.	103
6.3	The flow graph of our approach.	109
6.4	Ambiguity removing process in POS tagging approach.	115
6.5	Experimental framework of our POS tagging approach.	117
6.6	Sample result of splitting the input text.	117
6.7	Sample result of POS tagging.	118
6.8	Data set of different domains for testing.	119
6.9	Comparative analysis of precision, accuracy and sensitivity for various genre.	120
7.1	Experimental framework of our chunking approach.	132
7.2	Decision tree architecture for noun chunk identification.	135
7.3	Finite machine automaton for noun chunk identification.	138
7.4	Sample result of chunking.	141
7.5	Data set of different domains for testing.	141
7.6	Comparative analysis of precision, accuracy and sensitivity for various genre in chunking.	142
7.7	Per chunk tag error analysis.	142
8.1	A complete flow process of frequency generation of words in our corpus.	147
8.2	Scatter jitters of selected hundred words from different frequency bins.	152
8.3	Comparison of information scattering in (a) BNC corpus words scatter behavior; (b) Scatter jitters of selected hundred words from different frequency bins.	153
8.4	Measurement of level of information scattering for <i>DSHTC</i> Hindi corpus for the selected words Using the Juilland and Rosengrens methodology of dispersion measure.	154
8.5	Validation of words distribution in of the our corpus through Zipfs graph.	156
8.6	Sample result of line segmentation.	158
8.7	Sample result of word segmentation.	158

List of Tables

1.1	Some of most widely used corpus linguistic model.	5
1.2	Ground-truth of some well known handwritten corpus.	7
2.1	Standard corpus with their specification.	23
2.2	Handwritten corpus of English, Spanish, Indonesian, Korean and Chinese languages.	25
2.3	Handwritten corpus of Chinese, Tibetan, Arabic-Latin, Japanese, Greek, Urdu and Farsi languages.	26
2.4	Handwritten corpus of Arabic, Urdu, English, Japanese and Chinese languages.	28
2.5	Indic written corpora of 14 languages.	29
2.6	Indic handwritten corpus.	30
2.7	Devanagari Hindi handwritten corpus.	30
3.1	A hierarchy of linguistics analysis levels.	39
4.1	Official language with total speakers and their script.	51
4.2	Statistics of the demographic distribution of the corpus.	62
5.1	Meta information of an XML formatted file.	79
5.2	Comparative analysis of Devanagari script handwritten text corpus (DSHTC).	87
5.3	Comparison of software capabilities for information retrieval.	89
6.1	Comparative study of POS tagger for Indian languages.	98
6.2	Comparative study of POS tagger for Hindi language.	100
6.3	Comparative study of POS Tagger for other Languages.	101
6.4	The tag set for Hindi with 29 tags.	112
6.5	Patterns and with their tag.	112
6.6	Some examples of rules using prefixes.	113
6.7	Some examples of rules using suffixes.	113
6.8	Performance result of POS tagging approach in terms of P-Precision, A-Accuracy and S-Sensitivity.	118
6.9	Comparative study of our POS tagger with other Hindi taggers.	121
7.1	Comparative study of chunking for Indian languages.	129
7.2	Comparative study of chunking for Hindi languages.	130
7.3	Comparative study of chunking for other languages.	131

7.4	Chunk tag set used in the system.	134
7.5	Performance result of chunking approach in terms of P-Precision, A-Accuracy and S-Sensitivity.	140
7.6	Comparative study of chunking models with our chunker.	143
8.1	Comparative analysis of Entropy, H-maximum and Redundancy of Indic scripts corpus and our corpus.	149
8.2	Analysis of three approach used for testing our corpus for text line segmentation benchmarking.	157

Chapter 1

Introduction

The development of a comprehensive and standardized corpus with annotation (structural mark-up and linguistic annotation) is of great importance. In the last six decades, the corpus development methodology is the most growing and widely spread technology in the linguistic domain. As of the linguistic perspective, a corpus is defined as congeries of texts compiled systematically to provide a platform for various linguistic research.

The trustworthiness and usability of a corpus depend on the coverage of the optimal texts and quality of the texts culled in the corpus. The recent advancement of computer predicted compilation of corpora made it even more facile and opened many incipient areas for research in the Natural Language Processing (NLP).

1.1 Linguistics of Corpus

Conventionally, corpus can be defined as a large assortment of the machine readable text, that are used for the many linguistic researches. There exist two types of hypothesis findings using the corpus methodology, first one is the corpus-based approach and second is corpus-driven approach. The corpus-based approach conceived theory from the real data and its validation against the corpus data.

Corpus linguistic is the backbone of linguistic studies, it could be used to study about the many branches of the linguistics such as: syntax, phonetics, semantics, lexicographic and document image processing. The role of the corpus linguists grew noticeably in the past decade, due to massive explosion in the technology,

mainly in the high speed computer and efficient storage systems. The increasing demand of computer in the corpora, tend the linguists towards the corpus linguistics with a great interest.

The guideline principle for the corpus is not strictly definable, generally it can be defined as a collection of text, but these two terms are distinguished in linguistic field. Because, simple collection of texts without any guideline principle cannot be outlined as a corpus, for example the WWW (World Wide Web) is a huge collection of text but still cannot outline as a corpus. There is always a standard which should be followed during the corpus design and some principles are used during the information retrieval from the corpus.

The two linguistic terms are text and corpus. The text are always read in the horizontal manner while a corpus is read vertically, because the whole text of corpus cannot be useful for only one single perspective. The aim of corpus designing and development is always environed around the content and audience, while texts are used to deliver the particular ideas of operational and deliberation.

The text reading in a corpus specifies the phenomena about the language and data collection, the individual act of text. A particular event in corpus can repeatedly occur in different perspective while in the text it is unique. There is coherent communicative relation in text during the reading while corpus does not have coherent communicative relation in all texts due to random selection of texts.

To make the corpus more significant and envoy for various discipline of the linguistics, annotation of corpus and statistical profiling are the two foremost methodologies used to exploit any corpus data.

1.1.1 Annotation of Corpus

Annotation of corpus is a necessary task from linguistic perspective, it can act as a increasing demand of the corpus linguistic and provide a detailed analysis for the same linguistic level and support the research for the subsequent level. In recent years so much concentration has been given on the annotation of corpus. An annotation of corpus assigns an extra value (linguistic information) to a corpus that can be used in various applications of NLP and related to speech.

We have applied a novel approach for structural annotation of handwritten document at different levels such as: text-page, line and at word level. The approach

of annotation is not only limited with the text annotation, we have also done the textual region ground-truthing, part-of-speech tagging and chunking for the Devanagari Hindi text. By applying the same approach, we extend the functionality of model for the benchmarking and testing of various Optical Character Recognition (OCR) algorithms. With the involvement of computer and linguists the raw data is reformed with some additional paramount linguistic information to make it more representative in a systematic way to utilize it with machine readable operation.

1.1.2 Statistical Profiling

To find out the behavior of a particular text, frequency of occurrences of a word is the standard methodology or for some selected tokens that can be individual or group in a corpus. The frequency listing defines the record of how much times a word occur in a corpus data. The frequency list can be arranged in order with different terms like alphabetically, tagging or in a group.

One more type of frequency profiling is, to find out the concordance relation among more than one word. Interpretation of concordance lines are used to find out the information about the position of a particular word. For example find the number of occurrences a word at center, before and after a particular word or tag.

Frequency profiling is the only standard term utilized to characterize the congruity of a corpus as a linguistic resource. In this thesis, we have used the frequency profiling and delineate the scatter behavior of words distribution in our corpus. The statistical measurements of corpus information help to discover the isolation and similarity between two firmly related languages.

The main goal behind considering the statistical frequency profiling are as followings:

1. To test distributional behavior of data and to follow the standard linguistic rules for designing a standard corpus.
2. To test whether the frequency distribution of data in corpus is actually reflective of the sense of the linguists in which they are intrigued.
3. The statistical interpretation of frequency data is necessary, which determines the degree of statistical association with words.

4. To test hypothesis about a language and to explore grammatical information of a language to generate various mathematical aspects about the language.

1.2 Motivation

In recent era technology has got tremendous growth in digitization of information, but in India, each year a massive volume of the handwritten forms are digitized manually in both public and private organizations for communication and official work likes in banking system, railways reservation, post office, census calculation, examinations form processing, passport etc. It requires a lot of time, money and manpower for processing.

To decipher these issues, a standard database is required where different OCR techniques can be trained and tested for automation in handwritten form processing.

A wide range of the corpus tools or softwares are developed by linguists, to perform various linguistics operation on raw corpus. Most of the linguistic tools are limited with the functionality, they are operated either on the electronic or handwritten text corpus. Some of the well known tools developed for the compilation or statistical analysis of the handwritten or Unicode corpus are as shown in Table 1.1.

Among these tools, SARA, WordSmith and BNCWeb provide the functionality of the quantitative analysis of the corpus data, while GATE provides the functionality of corpus annotation and information retrieval. Besides this, Matrix provides the functionality of both quantitative and statistical analysis of corpus data and cross corpus comparison. These tools are detailed in later sections of this thesis.

Following are the linguistic tools developed for the handwritten corpus such as: GTLC [Yin et al. (2009)], APTI [Slimane et al. (2009)], PixLabeler [Saund et al. (2009)] and Truthing Tools [Elliman and Sherkat (2001)]. Among these tools, PixLabler and Truthing Tools provides an annotation for English language documents. APTI and GTLC provides an XML Ground-Truth (GT) for Arabic and Chinese handwritten documents respectively. GTLC developed the three levels GT for text-page, lines and words, while APTI is limited with text-page annotation only. However, again these tools are limited with handwritten corpus annotation only.

The above drawback motivated us to design a system, which could be compiled

Table 1.1: Some of most widely used corpus linguistic model.

Model	Unicode	Hand Written	Compi- lation	Stati- stics
SARA [G. and L. (1998)]	Yes	No	No	Yes
Word Smith [Lehmann and Hoffmann (2000)]	Yes	No	No	Yes
GATE [Cunningham et al. (1997)]	Yes	No	Yes	Yes
Matrix [Paul (2003)]	Yes	No	No	Yes
Tools [Elliman and Sherkat (2001)]	Yes	No	Yes	No
BNCWeb [bnc (2007)]	Yes	No	Yes	Yes
GTLC [Yin et al. (2009)]	No	Yes	Yes	No
APTI [Slimane et al. (2009)]	No	Yes	Yes	No
PixLabeler [Saund et al. (2009)]	No	Yes	Yes	No
Truthing Tools [Elliman and Sherkat (2001)]	No	Yes	Yes	No

and annotate, at structural and linguistic levels. With supporting handwritten document and associated transcript text file with counterpart platform. This could be helpful for the linguists to enhance most of linguistics research. In addition of these, the model also performs both quantitative and statistical analysis of corpus data to compute the linguistic information about the corpus. So that a mathematical aspects about the language and corpus are generated.

We have selected Devanagari script in Hindi as it is the fourth most frequently used script in the world as an input source for experimental setup of our model for development of a standard corpus. The Hindi language is considered for handwritten corpus development and statistical analysis because Hindi is the Indo-Aryan language and there are poor resources in the field of handwritten Hindi document ground-truth field. The main reason behind the lack of standard database for Hindi is the poor availability of resources for data aggregation and its adversity. These issues motivated us to develop Devanagari script handwritten text database that is a much-needed model for training, testing and benchmarking of handwritten text recognition tasks. We developed a perceptive machine vision system which automatically associates transcribed texts of handwritten Hindi typescript into their counterpart.

A variety of standard handwritten databases are developed for the scripts such as: Chinese HIT-MW database [Su et al. (2007)], Japanese ETL9 [Saito T (1985)], English [Marti and Bunke (2002)], FHT database for Farsi [Ziaratban et al. (2009)] and Arabic database [Lawgali and Bouridane (2013)]. PBOOK (Persian, Bangla, Oriya and Kannada) [ALAEI et al. (2012)] database is a multilingual handwritten database, developed for four scripts.

From the literature reviews, it is concluded that for Hindi language, there exist poor resources of handwritten databases.

For the exploration and research on a particular language a large volume of both handwritten and Unicode database are required. In this work, we describe our approach for design and development of a large volume of Hindi handwritten text images corpus along with the Unicode and structural annotations. To make the database available for large area of NLP, we annotate database at structural and linguistic levels for each image and associate an auto-generate XML based ground-truth meta information to make it computer compatible as a linguistic resource. Because, XML is the most demanding and suitable platform for the document image analysis. This work also addresses some challenges related with corpus design and annotation such as data collection, writers selection, methodology of

annotation etc.

There is a significant need of linguistic resources that train a system to extract and recognize text efficiently from handwritten documents. Annotation of textual document is time consuming and error prone procedure which requires utmost heed.

The ground-truth database is the basic need in document image analysis because, various parameter are associated with the term ground-truth which provide a standard platform for training and testing of various optical characters recognition techniques. In handwritten document the style of writings vary based on person, circumstance and environment. The standard English handwritten database IAM [Marti and Bunke (2002)] provides an XML ground-truth for the text-line and text-word, but the process of XML ground-truth is manual where the developer manually create an align XML transcription for the handwritten texts. Table 1.2 shows the ground-truth of some well known handwritten corpus.

Table 1.2: Ground-truth of some well known handwritten corpus.

Database	Level	Technique	Represent
GT(Bangala) [Sarkar et al. (2012)]	Text-line	Plain text	txt file
IAM-HistDB [Fischer et al. (2010)]	Text-line	Plain text	txt file
PBOK [ALAEI et al. (2012)]	Text-line	Binary	txt file
IAM [Marti and Bunke (2002)]	Text-line Text-word	XML	XML file
GTLC [Yin et al. (2009)]	Text-line Text-word	XML	XML file

In this work, we have designed and developed a Devanagari script handwritten corpus with structural and linguistic annotation for the compilation and statistical analysis of Hindi corpus data. The approach is also vigorous to some extent that, it consider both the data collection and annotation process in the account for the development of corpus in a systematic way using the particular framework. We specify the new criteria for data choosing and structural mark-up of corpus at different levels. We have also used a basic yet instinctive framework that can annotate a vast database and gives a more prominent effectiveness as compared to other corpus annotation tools. In addition to Hindi corpus compilation, we have described a variety of statistical analysis techniques with the corpus information to make it more representative for describing the behaviour of texts in corpus information. The statistical results also present the technical specification of the Devanagari script in Hindi for mathematical and computational aspects.

A statistical analysis is conducted based on fundamental statistics like word frequency distribution, the rank of the word, entropy and perplexity of corpus data, mean, standard deviation (σ) and variance of coefficient (C_v). Based on the results obtained from embryonic statistics, we have evaluated manifold statistical methods such as measures of dispersion with frequency profiling, [Juilland A.G. and C. (1970)] and [Rosengren (1971)].

1.3 Challenges

This work has been done to compile the Devanagari Hindi corpus and address the following challenges as:

1. In terms of linguistic we have added the features of tagging, chunking and structural ground-truthing of corpus text.
2. For the easy retrieval of desired information, we have developed a framework to store the text in systematic format and same framework can also be used for various NLP application to test their accuracy on benchmarked data set.
3. Collection of raw data from different genres (politics, sports, history etc.). Beside this, cater to maximum syntactic variations the writers are chosen from different age groups, profession, educational qualifications and geographical location of India are involved in handwritten data collection, both who are comfortable with Hindi and those whose mother tongue is not Hindi. The form filling sessions were carried out at distant locations like shopping malls, railway stations, bus stand and hospitals etc., as the handwriting of a person sometimes gets affected by the mood, situation and surroundings.
4. Statistical analysis of data set is done to compute frequency, entropy contingency and Zipfian distributions. Beside this, dispersion is measured using pure frequency profile and standard deviation.

1.4 Aims and Objectives

The aims and objectives of this research are to investigate and provide automation for the compilation of corpus data (both handwritten and transcript text),

linguistic annotation and statistical analysis of corpus to validate it as a standard platform. Our research objectives are divided into following parts:

1. To design and develop a standard Devanagari script corpus and supportive corpus linguistic framework.
2. To develop a perspicacious machine vision system to compile the corpus and associate the aligned transcript texts of handwritten Devanagari Hindi texts.
 - (a) To explore the linguistic features of compiled corpus data for further investigation on data.
 - (b) Provide a quantitative overview of the corpus data in various terms.
3. To perform extensive quantitative and statistical analysis to ascertain its applicability as a mathematical language model and to discover about the scatter behaviour of data inside corpus.
4. To provide a tiered structural textual region XML ground-truth and linguistic for the handwritten documents to support the benchmarking and training of various OCR techniques.
5. To provide POS tagging and chunking of Devanagari Hindi text to enhance all linguistic research need on same platform.
6. To provide an effective tool for learning basic grammar without any human tutor implication along with real life examples with augmentation capabilities.
7. To provide a platform which can be used for Information exchange and a Translation tool.

The goal of this exploration is to propound a framework that can be applied for the variety of linguistic research likes compilations of corpus, lexicography, annotation, grammatical tagging, benchmarking of OCR techniques and quantitative analysis of linguistic variables, content analysis, machine translation and learning and teaching a language. The output of results are represented in various format to serve different purposes such as an XML files, query based results for statistical analysis, filtering of requested contents with framework inbuilt functions etc.

1.5 Thesis Organization

The thesis is structured in the following manner.

Chapter 2 explain the fundamental philosophical phenomena about the corpus linguistic, where we have discussed the corpus linguistics with various terms and defined the need of corpus and evolution in the corpus field. Chapter describes the guidelines and principle which should be followed during the compilation of corpus, and different notion of texts selected from various types of sources. The chapter includes, the detail literature survey about the corpus linguistic in which we have focused mainly on the handwritten corpora. This leads to a consideration about the requirement of corpus in the linguistic domain and interest of linguists toward the corpus. The chapter also describes the current corpus based activities in various linguistic branches and how representative corpora can be used.

Chapter 3 discuss the widely used tools available for the linguistic annotation and text analysis. Chapter describes the details about various types of annotation standard used for both handwritten and machine readable texts in which we have also considered the technical issues occurring during the annotation process and query based information extraction. Chapter also explores the available corpus supportive linguistic framework in terms of information extraction and retrieval.

Chapter 4 describes the complete process of corpus compilation and methodology used for selection and balancing of text within corpus, based on various parameters. The chapter also provides the statistics overview of the corpus data and demographic information about the writers.

Chapter 5 presents our model and methodology used to implement this model. The chapter discusses the complete process of structural annotation and ground-truth of handwritten text-form. The chapter explains the standards used during the annotation process, auto-indexing approach and backhand database configuration. Chapter includes the application and functionality of the model. The chapter also explains comparative study of the model with existing handwritten annotation tools.

Chapter 6 details the part-of-speech tagging approach for Devanagari Hindi text. Our POS tagging method is performed with the help of probabilistic approach and rules-based approach. The method works in two phases, in first phase it tags known words according to trained data set and in the second phase it tags unknown words using rule based approach. All the rules are derived by calculating probabilities

of current word, previous word and next word in the database. An ambiguity removing module is also provided in the approach.

Chapter 7 presents an automatically chunking approach for Devanagari Hindi text. Our chunking method does not use any manual chunked corpus of Hindi language. Approach uses only the part-of-speech information of the text as contextual information. The approach is based on rule-based model. These rules are generated by detailed analysis of POS tagged corpus of Devanagari Hindi language and represented through regular expressions. Various FSM automaton are used to implement these rules.

Chapter 8 discusses the detailed statistics overview of the corpus data. The chapter presents an evaluation of our model and method from various statistical and benchmarking perspectives. Here, we evaluate the statistical results of the corpus data and then we validate these results using different statistical analysis techniques. The chapter describes the processes used to discover the various quantitative and statistical analysis done on the corpus data, to strengthen the claim and applicability of our framework as a standard linguistic resource.

Chapter 9 concludes the study and also discusses some restraints of the model and future work on the model and methodology.

Chapter 2

Literature Survey of Corpus Linguistic

In the resented era of information digitization, the corpus linguistic can be deliberated with the augmentation of computer processing and storage of ample size corpus. In NLP for development and analysis of various linguistics activities, design of corpus is prerequisite. An assortment of attempts were made to figure out the analysis of the corpus data and compare the performance results of various corpora using assorted approaches. The majority of the corpus linguistics tools focus on a petite case study rather than detail interpretation. Our perception is to find the differences between association measures on large amounts of data and authentication of data collection.

2.1 Introduction

This chapter presents and review, logical design and development of the corpus in explicit linguistic authenticity of corpora. In addition to these the characteristic of corpus approach and taxonomy of the corpora will be portrayed. The literature survey of the standard corpora is presented with their detail stipulation. Which focus on handwritten corpora and corpus linguistics as a pursuit. At last corpus related activities in various domain of linguistics such as NLP, education, learning and teaching a language, Document Image Analysis (DIA) are detailed.

2.2 Corpus Linguistics Perspective

The corpus linguistics is a fastest growing province in computational linguistic research. The purpose of design and development of corpus with annotation is not limited with a particular researchers community, corpus is the backbone for the interdisciplinary research domain. Corpora are the foremost demandable platform for investigation on a specific linguistic analysis.

2.2.1 Corpus Linguistic Evolution

The most recent corpus linguistic on the grounds begin since the era of structural linguistic concept in the USA in 1948s. That time linguistic professionals (Harris and Hill) influenced with the optimistic and behavioral view of the science, considered the corpus as the fundamental philosophy of linguistic. They explained the need of a huge collection of naturally occurring texts from the language to manage the present challenge regarding linguistic. As a result, there was interruption between corpus linguistic of that generation and a later variety of corpus linguistic.

[Quirk (1960)] planned in 1959, to present a corpus of both spoken and written British English names as the Survey of English Usage (SEU). That was the first research center of Europe established to carry out the research on corpus.

The succeeding generation of the corpus linguistic, after few years of SEU, Nelson Francis and Henry Kucera formed a gaggle for corpus linguistic at Brown University, and the serious deliberation of the group comes with an output of Brown Corpus [Francis (1992)]. Brown Corpus is the collection of a sample printed American English with the dominating power of the computer for digital use in 1961. It was the beginning of the golden era of computer based corpus, later in 1975, the arduous task of rendering machine-readable platformed resulting London-Lund-Corpus (LLC) [Svartvik (1990)] resource for the study of spoken English. With the increasing capacity of processing [McEnery and Wilson (1996)] developed a large vocabulary London/Oslo/Bergen abbreviated as LOB corpus, later in 1986 released in the tagged version of the original corpus [Geoffrey Leech and Hofland (1986)].

After the thirty years on account that 1961, corpus linguistic expanded its scope and domain, it has come to be a spine of NLP procedure. Corpus grew to become

a resource for systematic storage of information for analysis/testbed of many linguistic associated hypotheses. Later in 1989, Plamondon and Leedham explore the concept of handwritten corpus, the evolution of corpus linguistic involved the handwritten collection for providing the platform to exploit the technologies of handwritten document based textual processing.

2.2.2 Corpus Linguistic in Linguists View

The word corpus is derived from Latin word “body”. So corpus is a “body” of texts in general sense. In present time, it can be defined as a representative collection of large texts of a particular language, dialect or subset of a language able to represent the nature of language for linguistic analysis.

According to [McEnery and Wilson (2001)] there are three perspectives of corpus as follows:

1. A body of texts.
2. A body of machine readable text.
3. A finite collection of machine readable texts sampled to represent a language or varieties in a language.

The word corpus can be defined in several ways as follows:

1. A collection of linguistic data, either written texts or a transcription of recorded speech, which can be used as a starting-point of linguistic description [David Crystal (1991)].
2. A collection of texts, especially if complete and self-contained: the corpus of Anglo-Saxon verse. In linguistics and lexicography, a body of texts, utterances, or other specimens considered more or less representative of a language, and usually stored as an electronic database [McArthur (1992)].
3. Corpus contains a large representative collection of texts samples compatible with computer processing, available for operational research in various domain of linguistic with systematic arrangement of information [Dash (2005)].

2.3 Principle of Corpus Linguistic

The corpus linguistics consent to see, how the language is used today in the different contexts of a language. It is additionally enabling us to teach language more comfortably. According to [Douglas Biber and Reppen (1998)] the corpus approach is comprised of imperative characteristics:

1. It is empirical examining the specific patterns of language use in traditional texts.
2. It makes use of a tremendous and principled collection of ordinary texts as the basis for evaluation.
3. It relies on both quantitative and qualitative analytically methods.

The corpus linguistic methodology is not strictly definable, but is because of the properties it acquired over time. It should be well-designed and cautiously developed. The corpus principle depends heavily on the sense and apparent consideration of the individuals concerned.

2.3.1 Salient Features of Corpus

A corpus should have certain standard linguistic criteria and characteristics to make it more supportive for study of a language as follows:

1. **Quantity:** The conventional, query which is generally raised by corpus linguistics developer, is: how large a corpus do we need to develop?. Selection of few random variables from a body is not able to be outlined as the corpus. It will have to be large in size to quilting all the essential amount of information either in spoken or written forms. Size is nearly the sum of its components, which constituent its frame.
2. **Quality:** The quality of a corpus is particularly associated with authenticity. That means the choate texts must be collected from the real lifestyles examples of writing and speech. The guise of a linguist is very essential. The linguist has to assert that the data collection has been finished from the everyday lifestyles based discourse, as a substitute than synthetic prestige.

3. **Representation:** A canonical corpus include samples from a wide range of texts to achieve appropriate description of a language. It should be balanced to all areas of language use to represent maximum linguistic diversities.
4. **Simplicity:** A corpus should contain text in simple and apparent text format. It should include undeniable texts in a simple layout. It anticipates an unbroken string of characters without any extra linguistic understanding marked-up inside texts. Simple textual content is adverse to any style of annotation with various varieties of linguistic and non-linguistic consideration.
5. **Equality:** Selection of the texts in the corpus have to follow the equality of contents. However, there is a controversial dilemma and may not be adopted everywhere. Sampling model could exchange substantially to make a corpus extra representative and multi-disciplinary.
6. **Retrievability:** The arrangement of information, data, examples and references should be handy to access from the corpus by the end-users. This pays concentration to preserve methods of language information in the electronic layout in a computer. The current technological know-how makes it possible to generate corpus in computer systems.
7. **Verifiability:** Corpus will have to be open for any empirical verification. We can use data from corpus for any facts. This places corpus linguistics steps ahead of intuitive procedure to language study.
8. **Augmentation:** The contents of the corpus should be increased regularly. This will keep the corpus updated with the linguistic changes occurring in a language over time and also make it available for the purpose.
9. **Documentation:** Separate document only includes the header representing information of the text. It allows effective management and flexibility in the corpus, with a small amount of programming effort plain texts can be separated for distinctive ambition.

We have followed the above mentioned criteria in our corpus, to make it supportive for multi-disciplinary research on Devanagari script. In addition to these aspects, we have highly focused on the Representation, Quality, Equality and Retrievability. The Quality and Equality criteria is achieved by the selection of real life samples of texts from different domain to cover all aspects and diversity in the Hindi language. Although for the Representation and Retrievability, we have

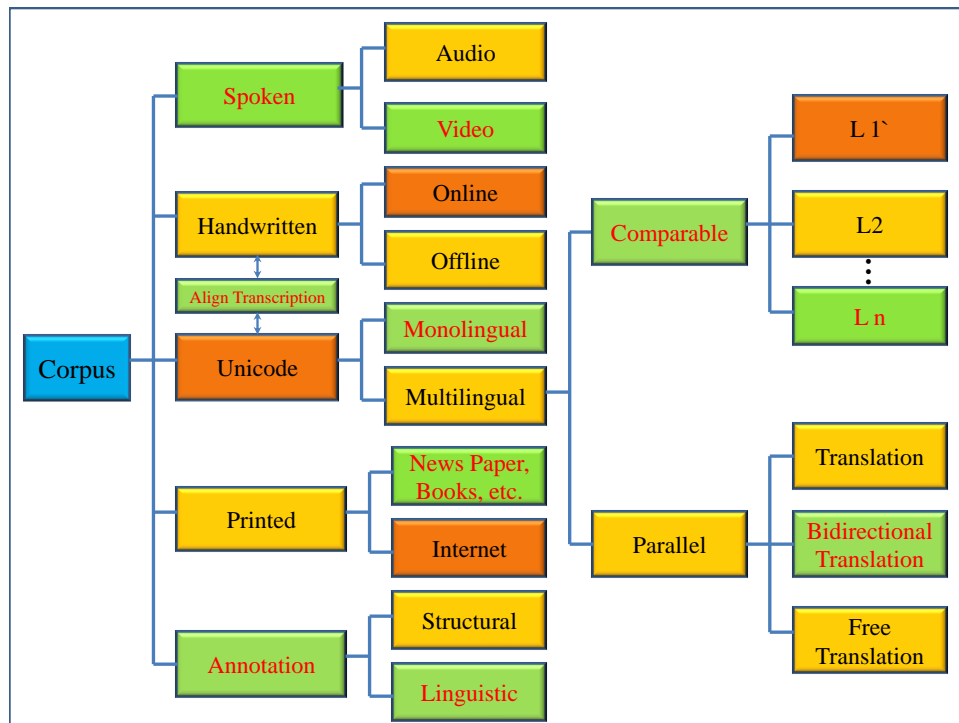


Figure 2.1: Conceptual classification of the corpora.

applied structural mark-up and linguistic annotation method on the corpus , including both grammatical and textual regions information to make it easy for the information extraction and analysis.

2.3.2 Taxonomy of Corpus

We distinguish the corpus linguistic based on the corpus design and contents, which will be helpful to understand the classification of corpus. The primary specification of corpus development building is related with two areas: the linguistic structure of design and conceptual classification of texts to select as sources. Figure 2.1 shows the categorization of the corpus, based on the corpus text.

2.3.2.1 Genre of Text

A corpus which is developed to constitute a representative sample of a defined language will be concerned with the random sampling of texts. Since the notion of texts is derived from the modes like electronic, written, spoken and speech conversion. Figure 2.2 shows the classifications of corpus according to various forms of text.

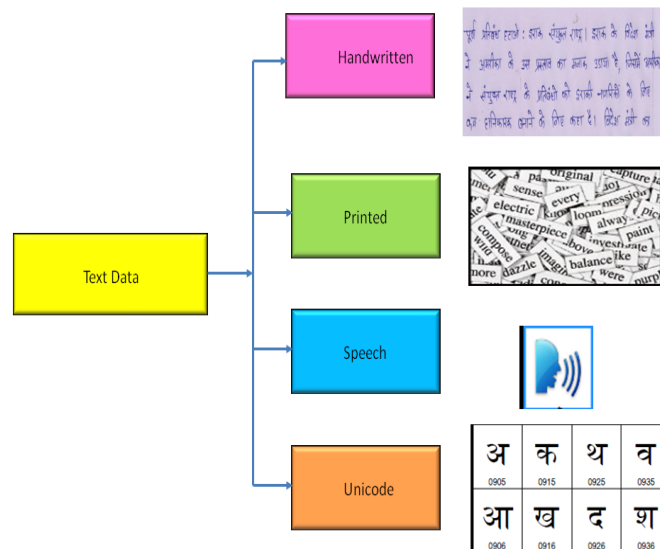


Figure 2.2: Text based classification of the corpora.

- **Electronic texts:** A corpus by morality of its genre contains only language data collected from various Unicode and electronic sources such as Brown Corpus [Francis (1992)].
- **Written texts:** A corpus where texts are often assumed to be a series of coherent sentences and paragraphs from various written, printed and published sources such as small ads in newspapers, an article from newspaper, magazine, poem, science etc., for example Bangla News Corpus [Majumder KMYA and M. (2006)]. Another category in written corpus is the collection of handwritten materials collected with involvement of various participants likes IAM [U.V. Marti (2002)], PBOK [Pal (2012)].
- **Speech corpus:** A speech corpus contains all formal and informal discussions, an informal face-to-face conversation, a telephone conversation, a lecture, a meeting, an interview, a debate etc., such as Wellington Corpus of Spoken New Zealand English [Janet Holmes (1998)].
- **Spoken corpus:** Spoken corpus is a technical extension of speech corpus, contains texts of spoken language such as London-Lund Corpus of Spoken English [Svartvik (1990)].

2.3.2.2 Nature of Text

Researchers developed a wide range of corpora, some of them were developed to carry out research on selected domain or to handle specific challenges. Certain linguistic criteria and characteristics based on these corpora are divided various categories as:

1. **General corpus:** General corpus includes the basic texts belonging to distinctive disciplines, genres and registers (e.g., British National Corpus) [BNC (2003)]. The corpus is dynamic according to nature it provides the opportunity to adopt newly available data over the time and continuously increments its vocabulary in size to extend the scope of utilization.
2. **Specialized corpus:** Special corpus is collection of a particular type of the text, to represent only a given type of text. It does not contribute to the nature of a language because the contents of the corpus is unbalanced, distorted and skewed. Its origin is not authenticated because the content of non-representative nature of the language involved. For example the corpus of language of children (CHILDES) [Hillsdale (1995)].
3. **Sub-language corpus:** It consists texts within selected domain of a particular language. American Nation Corpus (ANC) is the subset of Brown Corpus where the theme is same as Brown Corpus but the criteria of text selection has been reduced for ANC [Suderman (2004)]. MASC [Baker (2008)] sub-corpus of the ANC, is further derived from the ANC corpus and manually annotated.
4. **Sample corpus:** Sample corpus requires utmost care during the design and selection of contents because the corpus is static. Once the corpus is developed it does not allow any modification with the time and any other parameters to avoid the changes which make it unbalance and effect its constitution and research requirement.
5. **Historical corpus:** A special category of sample corpus is literary corpus. Such corpus include texts from different period (e.g., 16th century, e.g. the Helsinki corpus etc.), author, genre, group etc.
6. **Monitor corpus:** Monitor corpus is a growing, non-finite assortment of texts. Regular growth of corpus displays changes in language with new material, while old material can be removed from the corpus (this is not

strictly necessary due to incremented computer processing and storage). The aim of monitor corpus is to track and observe the long term variations in meaning, for example the Bank of English Corpus [Renouf (1987)].

2.3.2.3 Type of Text

Domain of the corpus boundary can be collection of texts from one particular language or more than one languages together for the comparison and translation. Corpus body can contains the mix texts of multiple language or translation form one language to others.

1. **Monolingual corpus:** It contains texts (written and spoken) of a single language representing its use in a particular period or in multiple periods. (e.g., ISI Bengali Corpus [ISI (1995)]).
2. **Bilingual corpus:** Bilingual corpus is formed when corpora of two related or non-related languages are put into one frame. (e.g., TDIL Bengali-Oriya Corpus [TDI]).
3. **Multilingual corpus:** Multilingual corpus contains representative collections from more than two languages (e.g., Crater Corpus). EMILLE is an example of the Multilingual corpus where a translation of English are available in 10 Indic scripts.

2.3.3 Representativeness

A corpus which has been developed to symbolize a language or a nature of language, can not predict that what type of question shall be made on it. So, developer should make this up it to interpret results of all queries properly consistent with criteria of purpose. The corpus representativeness is a crucial and essential part to set out the requirements. It should incorporate samples from the vast variety of vocabulary over a long period to contain the phrases lost with time. It must be balanced among the determination of phrases to represent maximum variance with linguistic standpoint.

Following are the steps towards achieving as representative as possible a corpus:

1. Decide the structural standards that are going to use to construct the corpus, and apply them to create a framework for the foremost corpus.

2. For each and every aspect draw up a complete stock of text forms which are found there, making use of best external standards.
3. Estimate a target measurement for every textual content form, referring to together, total target size for the aspect and quantity of text forms.
4. Record all these steps in order to end users can have a reference point if they get surprising results, and that enhancements can be made on the basis of expertise.

2.3.4 Purpose of Design

To make the corpus more useful, with the linguistic perspective the purpose of corpus design is classified in two broad contexts raw and annotated corpus:

1. Raw Corpus: This is the simplest form of the corpus, where text is collected in the natural form and stored without any modification. For example TDIL Corpus [Ganesan (2003)]. The raw corpus do not have any constraint for the arrangement of information.
2. Annotation: This process adds a linguistic information to a corpus, an annotation is closely related to corpus mark-up. Annotation is multi-functional and can be used for different purposes. Extraction of information (verb, noun etc.) from the annotated corpus is easy. Annotation of the corpus is done mostly at the word level called tokenization. Annotation also have many challenges as it is time-consuming, costly and requires the attention of a linguistic expert.

There can be following annotations:

(a) Structural Annotation:

Mark-up: A tiny amount of the extra textual information is added with the raw text. For example the information added with the audio recordings (data transcribed, speakers information, metadata) and extra textual information added with the written texts (e.g. title, chapter, author, year).

A document specific mark-up is as following:

- i. Document-wide mark-up: encoding, bibliographic description.

- ii. Structural mark-up: encodes figures, tables, headings, structural units of text, paragraphs etc.
- iii. Sub-paragraph structures: words, sentences, quotations, abbreviations, dates, names, terms etc.

Ground-Truth (GT): Contour information about the textual region is added with the document. The standards used for the corpus mark-up and ground-truth are defined.

(b) Linguistic Annotation:

In Natural Language Understanding (NLU) to extract information or answer questions that needs to recognize and understand text, linguistic annotation plays an important role in these processes. Linguistic annotation e.g. POS tagging, Chunking etc.

In our corpus DSHTC, the genre of texts we have selected for our corpus is, both handwritten and Unicode text to present most of the valuable demands of linguistics and language technology. To achieve the consistency between handwritten and Unicode text during the annotation process, we provide align transcription between handwritten and Unicode texts at different levels.

For annotation of a corpus to make it available in the wide domain of NLP, the raw texts of a corpus were reformed with some additional information as POS.

2.4 Survey of Corpora

The word corpus comes in existence in 1901. In the last decade of 20th century, various corpus of handwritten and printed text of different languages were developed. The corpus were developed with many purpose, but mainly the development criteria is classified in online or offline handwritten printed and handwritten corpus.

There are number of written and speech corpus available for English, Arabic, Chinese and Urdu etc. There is not much data set available for Hindi handwritten and printed text. Existing corpus focuses only on document specific characteristics and does not take into account the structure of the corpus.

Most of the corpus are designed and developed to meet the requirement of a specific research project. There are thousands of corpus in literature, an abundant number of the corpus for the languages other than English are also available.

Table 2.1 shows some well known large language corpus with their specification. British National Corpus (BNC) [BNC (2003)] is the first well known corpus for the English language. Corpus comprises of approximately a hundred million words of modern British English. The demographic aspect are as age group, sex, social type and geographical area. Later in 2007, BNC has released the XML version of the corpus [BNC (2007)].

The American National Corpus (ANC) initiated in 1998 and first time released online for uses in 2004 [Suderman (2004)]. ANC corpus was also used to generate the sub corpora, in 2008 based on this corpus a MASC [Baker (2008)], manually annotated corpus was developed. Developers also designed a web based framework for ANC [Suderman (2010)] the customization of the corpus. Corpus follows the theme of BNC with some difference regarding the sampling period of texts and expands the category of text selection. ANC choose the sampling period from 1990 onwards whereas the BNC sampling period is 1960 – 1993. Freiburg-Brown Corpus of American English (FROWN), LOB (Lancaster -Oslo-Bergen Corpus) etc.

Table 2.1: Standard corpus with their specification.

Corpus	Language	Description	Size (Million words)
BNC	B-English	Written and Spoken	100
WCNZE	NZ-English	Written	01
COCA	A-English	Written and Spoken	425
ANC	A-English	Written and Spoken	20
LOB	B-English	Written	01
Brown	A-English	Written	01
FLOB	B-English	Written	01
Frown	A-English	Written	01
Enron	English	Mail Messg.	0.2

2.4.1 Handwritten Corpus of English, Spanish, Indonesian, Korean and Chinese Languages

Some of the structurally annotated handwritten corpus with their specification like script, number of characters, words and sentences are shown in Table 2.2. The

most widely used handwritten corpus of English, Spanish, Indonesian, Korean and Chinese languages are shown in the Table 2.2. CEDAR [Hull (1994)] is image database for handwritten text recognition. In this the images are scanned at 300 dpi. These data are collected from a working post-office. The database is a collection of digital images of approximately 5000 state names, 5000 city names, 50000 alphanumeric characters and 10000 ZIP Codes. Data were extracted from the real mail samples. It has been very widely used in the experimentation of a wide number of OCR techniques.

[David and Alicia (2014)] developed BH2M a Barcelona historical marriage database with 174 handwritten pages and 56000 words by single writer. It is an image database of historical handwritten marriage records, collected from 15th century centralized marriage record register.

[Peb Ruswono Aryan (2011)] developed text database for Indonesian handwritten text recognition. The database consists 200 forms and each form have 20 pages. The database is collected from 200 writers. The form consists of handwritten texts as isolated digits, mixed digits, isolated upper and lower case letters, isolated lower and upper case words collected from college students.

[Kim and Park (1993)] developed Korean character image database known as PE92. In this, there are 23 sheets consisting 100 characters and a half sheet of 50 characters. The 70 sets are generated by 500 writers and 30 sets are by same persons.

[D. Llorens (2011)] developed handwritten character data set for Spanish known as UJIPenchars. The handwritten characters are collected on Toshiba M400 tablet PC.

[U.V. Marti (2002)] developed English handwritten database known as IAM data set. The data set is based on LOB corpus and used for extracting handwritten text and segment it into lines and words.

[Deng (2012)] developed NIST modified digit data set known as MNIST. It is a digit database developed from NIST database that is same as TIMT (Speech database created at Texas Instruments and MIT).

A Chinese database name as CASIA- ONHWDB, developed by [Da-Han Wang (2009)]. This is a publicly available database for research collected from university students. In this database, handwritten data are collected using Anoto pen. The

database consist 4037 categories of handwritten characters, in which 3866 Chinese character and 171 symbols, prepared by 420 writers.

Table 2.2: Handwritten corpus of English, Spanish, Indonesian, Korean and Chinese languages.

Database	Script	Description
CEDAR [Hull (1994)]	English	5000-City Names 5000-State Names 10,000-Zip Codes 50,000-Words
BH2M [David and Alicia (2014)]	Spanish	56,000-Words Single Writers
Indonesian [Peb Ruswono Aryan (2011)]	Indonesian	200-Forms of 20- Pages(Each form) 200-Writers
PE92 [Kim and Park (1993)]	Korean	23-Forms 100-Characters(Each F.) 500-Writers 12000-Total
UJIPencharas [D. Llorens (2011)]	Spanish	11,000-Characters 60-Writers
IAM [U.V. Marti (2002)]	English	1539- Pages 13,353-Lines 115320-Words 647 -Writers
MNIST [Deng (2012)]	English	70,000- Images of Digit
CASIA-OLHWDB [Da-Han Wang (2009)]	Chinese	1,694,741-Characters of 171-Symbols 420-Writers

2.4.2 Handwritten Corpus of Chinese, Tibetan, Arabic-Latin, Japanese, Greek, Urdu and Farsi Languages

The most widely used handwritten corpus of Chinese, Tibetan, Arabic-Latin, Japanese, Greek, Urdu and Farsi languages are shown in Table 2.3.

[Antony (2011b)] developed Chinese character data set named as HCL2000 with the help of 1000 different writers.

[Long (2011)] developed handwritten database for Tibetan languages, name as MRG-OHTC. [Njah (2012)] developed a Multi-language handwriting database MAYASTROUN, for Arabic and Latin languages. [Tomohisa (2014)] developed database for Japanese known as “Kondate”.

[E.Kavallieratou (2001)] developed Greek handwritten database of character, digits, and symbols with the help of 1000 writers of men and women of size 1760 forms known as GRUHD.

[Raza (2012)] developed an offline handwritten database for Urdu known as CENIP-UCCP. The database consist 400 digitized forms by 200 writers. [Nakagawa (2004)] developed Japanese handwritten data set. It is an online handwritten Japanese pattern character database.

Table 2.3: Handwritten corpus of Chinese, Tibetan, Arabic-Latin, Japanese, Greek, Urdu and Farsi languages.

Database	Script	Description
HCL-2000 [Antony (2011b)]	Chinese	3755-Characters 1000-Writers
MRG-OHTC [Long (2011)]	Tibetan	910-Character Classes 130-Writers
MAYASTROUN [Njah (2012)]	Arabic-Latin	67,825-Words,Character, Math Exp.,Signature 355-Writers
Kondate [Tomohisa (2014)]	Japanese	Text,Figure,Table, Maps
GRUHD [E.Kavallieratou (2001)]	Greek	102,692-Words 1000-Writers
CENIP-UCCP [Raza (2012)]	Urdu	2051-Lines 23,833-Words 200-Writers
Japan [Nakagawa (2004)]	Japanese	12,000-Characters 120-Writers 10000-Characters 163-Writers
FARSI [Hosseini (2007)]	Farsi	102,352-Digit Images 11,942-Writers

2.4.3 Handwritten Corpus of Arabic, Urdu, English, Japanese and Chinese Languages

Some of the structurally annotated handwritten corpus with their specification like script, number of characters, words and sentences are shown in the Table 2.4. The most widely used handwritten corpus of Arabic, Urdu, English, Japanese and Chinese languages are shown in Table 2.4.

[Pechwitz (2003)] developed IFN/ENIT database of handwritten Arabic words.

It is a handwritten offline database of Tunisian town names. Suen et al. [Malik (2009)] developed handwritten data set known as CENPARMI. It is an Urdu database of digits, numerals strings. Sabri et al. developed an offline handwritten database of Arabic text known as KHAT [Sabri (2012)] with the help of 1000 writers of different countries. It is an Arabic offline database store at 200, 300, and 600 dpi. The forms are collected from 18 different countries.

[Nawwaf (1999)] developed a compressive handwritten database of Arabic at Al-Isra University in Jordan. Which include 37000 words, 10000 numbers and 2500 signature of the Arabic language. It is a database of handwritten words, numerals and signature of Arabic, created at Al-Isra University, Jordan. Lawgali developed a handwritten database for Arabic named as HACDB [Hasan (2006)].

AHTID/MW [Slimane1 (2014)] is a handwritten database for Arabic. [Somaya (2004)] developed Arabic handwritten data set named as AHDB. RIMES [Grosicki (2009)] is mail message database of English collected from mails. It comprises of 12, 723 handwritten pages corresponding to 5,605 mails of two to three pages.

ETL9 [Velek (2002)] is an Electro technical laboratory database of Japanese language. It is a set of hand-printed characters in JIS Chinese with its analysis in Japanese.

HIT-MW [Su (2013)] is a Chinese handwriting database, developed by 853 handwritten forms and 186,444 characters produced under unconstrained conditions.

2.4.4 Indic Corpus Augmentation

Linguistic research over Indic corpus was started in 1981. India has a very complex and peculiar situation. According to linguistic, it is divided in four regions, Tibeto-Burman (1%), Austro-Asiatic (1.11% speakers), Dravidian (20.82% speakers) and Indo Aryan (76.87% speakers).

Constitutionally there are 22 scheduled languages out of which Hindi is the official and English is associated official language. The written corpora of 14 languages as shown in Table 2.5 are created under the project EMILLE (Enabling Minority Language Engineering). EMILLE [Baker (2002)] corpus has been developed in a collaboration among Lancaster University, UK, CIIL (Central Institute of Indian Languages) Mysore, IIT (Indian Institute of Technology) Delhi, IIALS (Indian

Table 2.4: Handwritten corpus of Arabic, Urdu, English, Japanese and Chinese languages.

Database	Script	Description
IFN/ENIT [Pechwitz (2003)]	Arabic	26,459-words 411-Writers
CENPARMI [Malik (2009)]	Urdu	60,320-Digits 14,890-Letters 19432-Words
KHAT [Sabri (2012)]	Arabic	1000-Forms 2000-Paragraphs
Al-Isra [Nawwaf (1999)]	Arabic	37,000-Words 10000-Digits 2500-Signature 500-Sentences
HACDB [Hasan (2006)]	Arabic	6600-Characters 50-Writers
AHTID/MW [Slimane1 (2014)]	Arabic	3710-lines 22896-Words 53-Writers
AHDB [Somaya (2004)]	Arabic	28,800-Words 300-Writers
RIMES [Grosicki (2009)]	English	12,723-Mail Sample 250,000-Words
ETL-9 [Velek (2002)]	Japanese	607,200-Characters 4000-Writers
HIT-MW [Su (2013)]	Chinese	853-Forms(4-5 Lines) 186,444-Characters

Institute of Applied Language Science) Bhubaneswar, and AMU (Aligarh Muslim University) Aligarh India. The CFILT English-Hindi corpus contains parallel corpus for English-Hindi.

EMILLE corpus is openly available for the non-commercial research purpose, corpus was composed of three formats monolingual, parallel and annotated corpora. Monolingual corpora include approximately 92 million words. The parallel corpus contains 200,000 English words and their translation in Bengali, Gujarati, Hindi, Punjabi and Urdu. The EMILLE Corpus is annotated with Character Encoded Scheme (CES) complaint of SGML in an Unicode format.

Table 2.5: Indic written corpora of 14 languages.

Language	Size (Million words)
Hindi	8.8
Assamese	2.6
Bangla	5.4
Kannada	2.2
Gujarati	7.8
Kashmiri	2.3
Malayalam	2.3
Marathi	2.2
Oriya	2.7
Panjabi	04
Sinhalese	4.9
Tamil	10.1
Telegu	04
Urdu	1.6

2.4.5 Indic Handwritten Corpus

Indic handwritten corpus are shown in Table 2.6. [Sarkar (2012)] developed handwritten database for Bangla and Bangla text with English. In this data set total 150 forms are collected with the help of 40 writers, out of 150 forms, 100 of pure Bangla and 50 of English and Bangla mixed named as CEMATERdb. The data set developed at Center for Microprocessor Application for Training Education and Research.

[Chaudhuri] developed a data set BANGLA. It is a handwritten on-line and off-line data set for Bangla numerals, with the help of 1212 tablet of Genius New Sketch for online data collection.

[Thadchanamoorthy (2013)] developed handwritten city name database for Tamil. The data set consists 256 city names.

[Pal (2012)] introduced PBOK a database and ground-truth for four different scripts: Bangla, Kannada, Persian and Oriya. Persian part of the PBOK data set contains 140 text-pages of three different categories (school dictation, general texts, historical) written by 47 Persian native speakers. The database contains 1787 handwritten Persian text-lines, 27,073 words/sub-words and 106,643 characters.

Table 2.6: Indic handwritten corpus.

Name	Language	Description
CMATERdb [Sarkar (2012)]	Bangla-English	150-Forms 290-Lines 18,750-Words 40-Writers
BANGLA [Chaudhuri]	Bangla	8348-Online 23392-Offline Numerals
Tamil [Thadchanamoorthy (2013)]	Tamil	265-City Names 100-Samples 500-Writers
PBOK [Pal (2012)]	Bangla Oriya Kannada Persian	707-Text Pages 104,541-Words 12565-Text Lines 423980-Characters

Table 2.7: Devanagari Hindi handwritten corpus.

Name	Language	Description
Devanagari Bangla [Bhattachara and Chaudhuri (2009)]	Hindi Bangla	22,556-Devanagari 23,392-Bangla Numerals
Devanagari [Dongre and H.Mankar (2012)]	Hindi	1800-Sample 25-Writers
Indic [Bhattacharya and Chaudhuri (2005)]	Hindi Oriya Bangla	22556-Hindi 5970-Oriya 12,938-Bangla Numerals
Devanagari [Jayade- van (2011)]	Hindi	26,720-Legal Amount Words

2.4.6 Devanagari Handwritten Corpus

Devanagari handwritten corpus are shown in the Table 2.7. [Bhattachara and Chaudhuri (2009)] developed a mixed numerals handwritten databases of Indian scripts. The database includes 22,556 and 23,392 isolated handwritten numeral samples for Devanagari and Bangla scripts respectively taken from real-life situations.

[Dongre and H.Mankar (2012)] developed database for Devanagari characters, in which 1800 samples by 25 writers designed a platform of data exchange and

recognizer benchmarking.

[Bhattacharya and Chaudhuri (2005)] developed Indian scripts digit databases for Bangla, Oriya and Devanagari. The data are collected as for Bangla 12,938 digits by 556 writers, for Oriya 5970 digits by 356 writers and for Devanagari 22,556 digits by 1049 writers.

[Jayadevan (2011)] developed a Devanagari legal amount word handwritten database of Hindi and Marathi.

Existing corpora [BNC (2003)], [BNC (2007)], [Suderman (2004)] etc., of different interest, mainly focus on electronic corpus instead of the handwritten corpus without linguistic annotation. The existing handwritten corpora as explained in section 2.4.1, 2.4.2, 2.4.3, 2.4.4 and 2.4.5 are very limited, because annotation of handwritten corpora require exhaustive care as compared to electronic corpora.

Existing Devanagari script Hindi corpus such as [Dongre and H.Mankar (2012)], [Bhattacharya and Chaudhuri (2005)], [Jayadevan (2011)] etc. focus only on numerals and characters.

To address the absence or (to overcome the challenges) of a standard data set with large number of vocabulary and sentences from various categories to cover entire aspect of the Devanagari script Hindi corpus with structural mark-up and linguistic annotation, we designed and developed a corpus in both Unicode and handwritten format for Hindi language in Devanagari script.

2.5 Summary

In this chapter, initially we described salient features of the corpus methodology, which should be properly followed during the design process of corpus. Then a literature survey of the corpus linguistic is detailed specifically for the handwritten corpora.

We have surveyed the various existing corpora of different interest, our review of study concludes that, main focus is always given to the electronic corpus instead of the handwritten corpus without linguistic annotation, because of the development of a handwritten corpus requires a huge labour work. The number of existing frameworks for handwritten corpora is very limited, as annotation of handwritten corpora require exhaustive care as compared to electronic corpora.

To conquer these issues and high demand, we designed and developed a corpus in both Unicode and handwritten format for Hindi language in Devanagari script with annotation (Structural mark-up and Linguistic annotation). This method is detailed in subsequent chapters with database analysis.

Chapter 3

Corpus Compilation and Annotation Tools

Corpus linguistics is a approach that can be applied in many exclusive branches of linguistics to study and analyse linguistic such as pragmatics, semantics, syntax, morphological and phonetics. These are software models which are used for compilation and to explore corpus linguistic. These are interdependent and designed to incorporate the linguistic theory in a different way. In this chapter existing softwares models for corpus linguistic with the standard of achievements regarding intelligent, comprehensive modeling of a language are described.

3.1 Introduction

The stages required for corpus design and exploration can be classified into three steps: Corpus development (Selection, Insertion, Annotation and Encoding of Text), Corpus editing (Correction and Disambiguation) and Retrieval of information (Frequency Analysis, Statically Analysis, Concordance and Word sense) on different tools with various terms.

In this chapter, we will abridge the corpus linguistics supporting software or models and necessary terms regarding corpus annotation and editing to make it more representative for a particular purpose. On the other hand, we provide an outline and a reference framework for understanding the performance of corpus linguistic tools annotation.

This chapter describes the need for annotation and different levels of annotation used for the corpus, Chapter also presents the capabilities of various annotation tools developed for the processing of corpus-based information. Some of the tools are developed for the cross corpora comparison while some are dedicated towards the statistical analysis of corpus information. We also detail the need and process of ground-truth in document image analysis.

3.2 Linguistic Annotation

Design and development of corpus is a tedious and monotonous job, that requires expertise and also financial support. At the designing time, linguist always try to make the corpora more efficient and relevant in the long term usability by adding some interpretative structural and linguistic information (Part-of-Speech tagging and Chunking) to a raw corpus. These are detailed in the below section.

3.2.1 Annotation of Corpus

Corpus annotation is the scheme of associating some interpretative linguistic information to enhance the value of corpus in linguistic research. The example of an annotation is the adding of grammatical information with a word such as tags, label to indicating the category of the particular word. It also, describe the use of that word (Homonyms) in a particular sentence for example the word *rose* in sentences can have a different category with the different meaning likes: noun, verb, an adjective with meaning *flower*, my favorite flower is a *rose* and *to have gotten up* (he quickly rose from his seat), respectively. Therefore, these words are annotated with the different tags as per use of the word in the whole sentence.

Raw corpus are annotated by one or more annotators by different approaches using three types of annotation models such as manually, semi-automatic or automatic. Annotated information can be represented in various ways like SGML, XML, hierarchical tags etc. The process of annotation is an iterative cycle, in which the annotator identifies the standards, requirements and application domain, and designs an annotation model to apply on the raw corpus. The process of annotation is shown in Figure 3.1, as an iterative cycle, where the annotator identifies the standards, requirements and domain of application. Based on these an annotation model is built and applied on the raw corpus. The model is evaluated and validated upto benchmark to meet the purpose of annotation criteria.

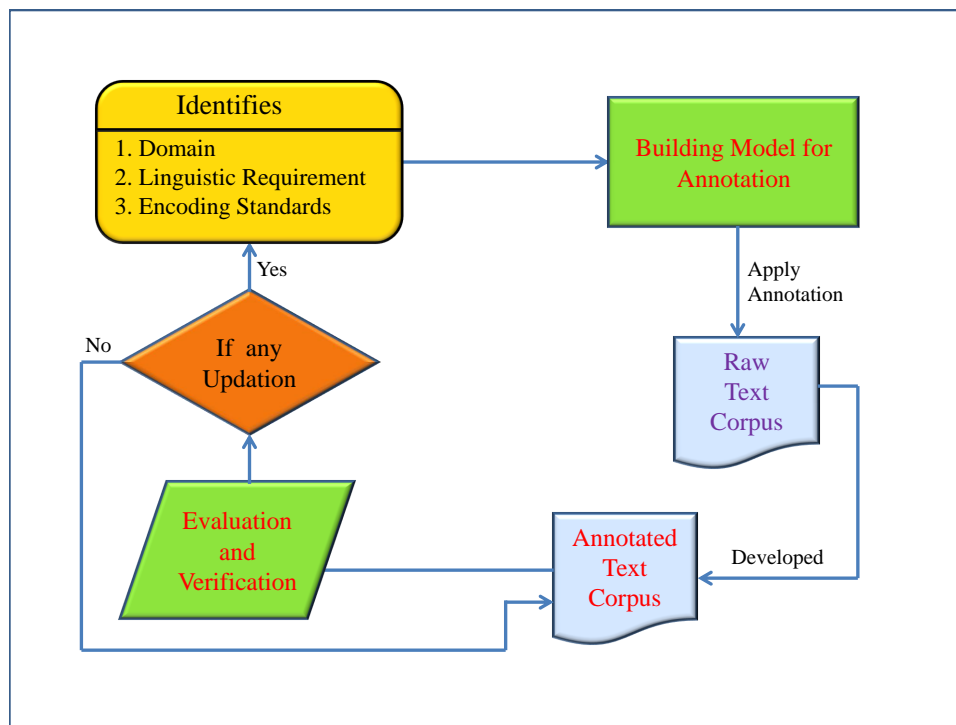


Figure 3.1: Process for generating an annotation model.

Annotation of corpus are not preferred every where, if the annotation is required then, the linguist should present and describe the intuition for annotating corpus [Sinclair (2004)]. The corpus without any alteration are required in the original format of plain texts, the reason given behind this is the error of annotation, and manipulation of originality. Alternatively, a majority view is giving more priority to the annotation of the corpus with a statement that annotation of corpus makes it more valuable for a broad range of linguistic purposes. Such as, the tagged version of the LOB [McEnery and Wilson (1996)] corpus of English language are serving the world for the various useful tagging research and is found more useful as compared to the original version of the raw corpus.

As we have discussed the annotation of plain text to corpora enriches it in terms of retrieving and analyzing information contained in the corpus quickly and easily. Adding semantic tags to corpus enhances it, for the utilization of the corpus in developing dictionary, and study of a language etc. Besides these, it also aids in parallel translation and transcription of the corpus, in the manual correction of some errors and in word processing software development.

Some advantages of an annotated corpus as standard practice of a good corpora are multi-functionality and automatic analysis of a corpus, such as:

1. Multi-functionality: Most of the annotation provided by the linguists are for

different purposes or applications. The linguists who build corpora predicted that the future uses of corpora are more variable than the originator of the corpus.

2. **Automatic analysis:** A corpus annotated in an advanced way, can be used in divers kinds of automatic processing or analysis of corpus. Such as, a corpus having grammatical tagging can automatically produce frequency lists with grammatical classification. It is also capable of generating the frequency dictionaries of corpus data with synonyms and antonyms, if these were provided during the annotation along with plain texts. In the case of the POS tagging, it can also classify the texts based on the word's pronunciation.

3.2.2 Taxonomy of Annotation

The linguists provide different types of annotation on the basis of diversity and task-specificity of corpora. To handle specific challenges or for the generalization purposes in corpus linguistic. Some of the standard annotations are as follows:

1. **Phonetic annotation:** It shows how a word is pronounced in a spoken corpus. Annotation of non-native speech is a cumbersome and extremely tedious task that requires an expert involvement for finding a proper balance between the expected benefit and the resources needed with acceptable compromise [Bonaventura et al. (2000)]. A phonetic annotation corpus of the 73,227 running monologues and dialogues words, with 9 hours and 46 minutes of correlative speech are developed by [Grnnum (2009)] .
2. **Syntactic annotation:** It is required for the creation, training and evaluation of parser and various language technological software tools to grammar check, information retrievers, and machine transcriptions or translators. Information is added about the parsing of a sentence in terms of syntactic analysis of units of phrases and clauses. A syntactic annotated version of the google book corpus is developed by [Yuri Lin and Petrov (2012)] to discover the uses of words and phrases over the time.
3. **Semantic annotation:** In this method of annotation the information about the semantic category of words are added. It can be performed in different ways like information extraction using grammar rules or by recognizing concepts and instances of structured texts. Example of semantically annotated

corpora such as PropBank [Martha Palmer and Gildea (2005)] and [Valerio Basile and Venhuizen (2012)] developed a large size machine based semantically annotated corpus.

4. **Pragmatic annotation:** It is useful to systematically identify pragmatic meaning in spoken text. That includes information about the sorts of discourse act. Occuring in a spoken dialogue. For pragmatic annotation, the two most biggest corpora with anaphoric and coreferential annotation are created at the University of Wolverhampton for English [Ruslan Mitkov and Sotirova (2004)], that contains 60,000 words, and for French [Agnis Tutin and Orasan (2004)] that contains 25,000 words.
5. **Discourse annotation:** Is related to information between sentences. Generally, there are two ways of specifying discourse relations, such as Informational discourse relations and intentional discourse relations. An example of discourse annotation are Pen Treebank (PTB) [Marcus and Marcinkiewicz (1993)] corpus of one million words.
6. **Stylistic annotation:** Stylistics annotation presents information addition for the field level inquiry, in which the associated information about the speech and thought presentation is used to insight and analyze linguistic theory. A significant way of stylistics annotation is defined by [Grice and Wilson (2000)], they have applied the systems of categorisation and analysis of linguistic science. Various stylistic approaches for the corpus annotation [Wynne (2005)] are presented using the XML and TEI. Marie et al. [Marie Garnier and Saint-Dizier (2009)] describes the stylistic guidelines based method for error detection and correction in annotated corpus.
7. **Lexical annotation:** An effective method to solve the ambiguity problems is a lexical annotation, it adds a lemma and tag to a word in a text. It also supports the morphological annotation, comparison of different annotation strategies for same text, searching for the particular word, tag, sense inventory etc. [Sorrentino et al. (2012)] proposed a supervised method for the lexical annotation of corpus.
8. **Ground-Truth annotation:** Besides the above mentioned annotations, one of the most effective category for the handwritten document annotations are “Ground-Truth annotation”, it is used to provide the most essential platform for the OCR algorithms.

The GT, imply to the precision of the training set's classification for supervised learning techniques. It is the statistical method to test research hypotheses. It is used in many regions for the supervised learning, detection of objects in video, remote sensing etc.

In GT, the algorithms are evaluated against ground-truth to test the hypothesis about the postulation. In addition to the benchmarking GT is also useful for successful testbed and training of findings. In handwritten documents the GT may allude a process in which a pixel on a textual region is compared to what is there in reality in order to verify pixel contents on the image.

The most widely used handwritten database such as PBOK [ALAEI et al. (2012)], IAM [Marti and Bunke (2002)], GTLC [Yin et al. (2009)], GT [Sarkar et al. (2012)] and IAM-HistDB [Fischer et al. (2010)] provides the handwritten images and corresponding ground-truth. [Gatos et al. (2014)] developed a cost effective semi-supervised procedure for the text-line level GT of historical handwritten document images.

The four scripts handwritten database PBOK, presents a manual ground-truth of handwritten document for the text-line level using the binary value 0 or 1 for the alternate line in text-page. The English handwritten database *IAM* provides a manually aligned XML GT for both text-line and text-word of the image.

The Chinese database GTLC provides the GT for both text-line and word of Chinese handwritten document image.

CMATER [Sarkar and Basu (2012)] database presents a text-line level GT, having both Bangla and Bangla-English mix handwritten documents.

IAM-HistDB [Fischer et al. (2010)] database presents the text-line GT for the historical handwritten document written in German script.

We have developed Devanagari script handwritten text corpus (DSHTC) with GT and linguistic annotation of Hindi handwritten text images. Besides these a textual region based pixel ground-truth of text-page and corresponding alignment between the text transcription are also done. The detailed methodology of the ground-truth implementation is described in Chapter 5. The linguistic annotation POS tagging and Chunking are detailed in chapter 6 and 7 respectively.

3.2.3 Hierarchy of Linguistic Annotation

Corpus linguistic annotation models are usually classified according to their functionality regarding corpus linguistics, and different tools may perform different analysis over various units. The tools are often synergetic, the results of the analysis performed at one level by the tool are, often used as an input for the analysis at a next level. The hierarchy of the linguistic analysis according to level is depicted in Table 3.1.

Table 3.1: A hierarchy of linguistics analysis levels.

Discipline	Units and Categories	Tools
Pragmatics Discourse theory Rhetoric Speech act theory	Discourse type Category Speech act category Emotion	Emotion analyzers Rhetorical coherency tools Named entity recognizers
Semantics	Predicate Logical representation Word sense	Word sense disambiguates Semantic role analyzers
Morphology lexical analysis Partial parsing	Word Prefix and suffix Grammatical gender Grammatical number Conjugation Phrases	Tokenizers Stemmers Lemmers Part-of-speech taggers Chunker
Phonetics and phonology	Sound Phoneme Syllable	Speech recognition tools Spectrograms/Sonograms Speech segmentation tools

However, the hierarchical dependencies between stratum neither provide any importance of particular stratum nor any implication representation about unique linguistic tools. It contains only theories for understanding the dependencies between levels, where higher levels generally depend on the lower ones.

For illustration of the semantic evaluation, before the parsing of the words, part-of-speech tagging or morphological analysis should be performed first then it will be feasible to clearly characterize the phrase. This determines that, the sentence must be clearly separated and tokenized from each other with the morphological analysed before parsing.

For illustration, the Penn Treebank [Marcus and Marcinkiewicz (1993)] and the British National Corpus [BNC (2007)] use different encoding standard and tag

sets. Various linguistic tools have different types of input and output depending upon the purpose and practice of their developer.

The corpora and computation linguists generally deal with the written corpus, because the written text often does not require any phonetics analysis and the finding of written text is also very easy in large quantities. Here, in next section we will detail the tools which are dealing with the written text corpora in specified data storage formats.

3.3 Corpus Annotation Standards

As discussed previously that annotation adds more value to the corpus, the annotation of plain text corpora enhances it in terms of retrieving and analyzing information contained in the corpus easily and quickly. Such as adding of grammatical tags to corpus is improving it for the utilization in developing dictionary, learning and teaching a language etc. Besides this it is also beneficial for developing a model for parallel translation and transcribing of the corpus, manual correction of errors and word processing tools.

Therefore, the implication of an annotated corpus depends upon the standards followed during the annotation process. The annotation system emphasizes the need for detailed hierarchical planning, so that prominent research results could be achieved.

3.3.1 Annotations should be Separable

Annotation is always added as optional extra information to the raw corpus, it should be associated in a specific way as per requirement to the plain texts and so that annotation can be separated easily with adequate formulation. So that the raw corpus can be fetched in its exact primitive form. This will be beneficial for the users who do not conveniently find annotation useful or want to annotate in other way with a particular goal.

3.3.2 Explicit Documentation

The annotation process should have a detailed explanation of annotation in well documented format with the useful information likes purpose of annotation, which

standard techniques were used for annotation, what coding scheme was used, accuracy rate of annotation. The document should portray the details about the unified set of categories used for the annotation, because mostly we observe that some people can more or less concede on the annotation.

The European Union EAGLES (Expert Advisory Groups on Language Engineering Standards) [Nicoletta Calzolari (1996)] started an initiative of the standardisation of NLP work in academic and industry. One of the subgroups of EAGLES union started to work on the standardisation of the corpora, the results come with various document specifying guidelines for corpus annotation

Annotation of a corpus usually means to enhance the value of raw corpus by associating the various type of linguistic information. This is done by attaching some kind of information as a parts of the data with respect to a particular understanding. Annotation of raw corpus can be done by one or more number of annotator by an automatic annotation tool or manually or semi-automatic tools.

3.3.3 Encoding of Annotation

The annotation encoding is attachment of grammatical tag, for example a word from Brown Corpus [BNC (2007)] *CAIRO* and tag *NP1* with tag description as singular proper noun, this indicates the tag itself can be simple or entangled. In the encoding of corpus annotation and representation a prodigious growth has been achieved in early 1990s by the utilization of standard encoding or mark-up language such as HTML, SGML and XML. The advantage of using these languages for encoding is, these languages are platform independent and developed with a world wide web standard. A set of tools developed at the Human Communication Research Centre (HCRC) such as *NITEXML* toolkit [Carletta J. (2005)].

The standard set of rules are required for development of interoperability tools and sharing or exchange of data between various projects. In recent era various standards set for this aspiration are as follows: XCES (XML based Character Encoding Standard) [Romary (2003)] [Romary (2004)], work within ISO TC 37 / SC 4, TIGER-XML and PAULA [Przepirkowski and Banski (2011)], and TEI (Text Encoding Initiative) [Sthrenberg (2012)].

The ISO (International Standard Origination) formed a Technical Committee 37/ Sub-Committee 4 (TC37 / SC4) in early 2000s to work in this area, working group

give some set of standard rules based on these results they set up two current projects CLARIN and FLaReNet [CLARIN (2009)].

3.3.3.1 Text Encoding Initiative

The Text Encoding Initiative (TEI) was a consortium established in 1987s with the involvement of academic, institution, research scholar etc. TEI develops and maintains standard supporting tools to represent texts in digital format. British National Corpus [BNC (2007)], CELT Project (The Corpus of Electronic Texts) and the Oslo Multilingual Corpus [Johansson and Hansen (2008)], are some of well-known examples of TEI based development.

3.3.3.2 Corpus Encoding Standard with XML

Multilingual Text Tools and Corpora (MULTEXT), along with collaboration of EAGLES and VASSAR developed a Corpus Encoding Standard (CES) that was used in a wide range of corpus encoding. That defined a minimal set of rules that each corpus should adopt for consideration of the corpus as a standard corpus. CES is also available in SGML and XML.

Various resources and projects using the XCES, for example National Corpus of Polish have XCES based encoding. XCES using the XML based Document Type Definitions (DTD) schema.

In our, handwritten corpus of Devanagari script handwritten text corpus with annotation (DSHTC), we have used the XCES (XML Character Encoding Standard) and TEI (Text Encoding Initiative) technique for encoding and transcribing the corpus data in an XML hierarchical way. In the representation of encoding data, we have also used an XML-based DTDs schema under the guidelines of TEI.

3.3.3.3 ISO Technical Committee-37/ Sub-committee-4

An ISO standard of Technical Committee-37/ Sub-Committee-4 (TC37/SC4) are developed as follows:

1. Designing of new work proposal.
2. Preparation of drafts documentation.

3. Acceptance of the documentation by committee.
4. Standard documentation distribution among the ISO committee members for review and voting.
5. Approval of the final documentation draft after voting pass as an International Standard.
6. Publication of International Standard for the implementation in work.

3.4 Corpus Development and Analysis Models

In the process of corpus data collection and development, specially designed software tools are required to view and analyse the corpus data. To develop a written corpus a set of hardware and software are required such as document scanner, OCR software, digital ink pen. In the spoken corpus, the hardware like speech recording, transcribing machine are required. Here, in subsequent sections we have described the existing corpus supportive frameworks that are needed in corpus compilation or annotation and retrieval of information.

3.4.1 Automatic and Manual Analysis

An annotation of the corpus make it more relevant. So the analysis of compiled corpus or annotation is performed in either of the two obvious ways such as automatic analysis and manual analysis. These can be further extended to semi-automatic, in which a human interpretation is also involved to increase the efficiency of the annotation process.

In automatic annotation, a detailed information with the original contents are added with the help of software. The process runs without any human interruption. The human intervention is limited with the start of execution and in defining a standard set of parameters. The majority of automatic annotation are trained based on the manually annotated samples which are based on the statistical machine learning approach or some set of rules. The process cannot be fully accurate in the annotation.

In comparison to automated annotation researchers always prefer the manually annotation corpus because the manually annotated corpus might be considered

as correct with linguist view. The possibility of errors are random and unbiased compared to errors made by the software. There is no other surrogate to construct a correct annotated corpus other than the manual involvement of highly trained linguists experts. Despite this, the manual annotation of the corpus are always expensive and smaller in the size in comparison to automated annotated corpus, but the development of manual annotation is essential and pre-required task for training and development of automated annotation tools. The tools of current time are trying to overcome above issues by concerting on a semi-automatic annotation tools, where annotation is dealt in automatic with manual correction of errors. Various tools which annotate linguistic data are described in the following section:

3.4.1.1 Prevail for Annotation

To represent the linguistic annotations of time series data, AGT (Annotation Graph toolkit) [Kazuaki Maeda and Lee (2002)] framework is developed. The framework annotation graph are independent from the file formats, coding schemes and user interface. That provide a logical layer for annotation system. The applications of framework are Multi Trans, Table Trans, TreeTrans and InterTrans.

Architecture and Tools for Linguistic Analysis Systems (ATLAS) [Christophe Laprun and Pajot (2002)] is a framework that provides an architecture to facilitate the development of linguistic annotations. It is an initiative issued with the involvement of NIST, LDC and MITRE. It presents a complex annotation for the arbitrary dimensionality signals for text and audio.

The annotation view at ATLAS core gives the deliberations on which whatever remains of the system is fabricated. These deliberations can be executed utilizing assorted programming languages. NIST has made a Java instantiate of the data model and gives an Application Programming Interface (API) to the core items permitting their simple control.

In addition to these, semantic information communicated utilizing ATLAS deliberations can be serialized to XML utilizing the ATLAS Interchange Format (AIF) to encourage their trade and reuse. Presently, an important dimension has been recently added to ATLAS: the Meta-annotation idea permitting compelling of the bland deliberations for particular needs utilizing the Meta-Annotation Infrastructure for ATLAS.

3.4.1.2 Prevail for Annotation and Information Extraction

An integrated set of XML tools is LT XML [Chris Brew]. It is an application oriented framework that provides facilities such as text processing in XML documents, text annotation through mark-up architectures, searching and extracting, report generation and formatting, tokenising and sorting.

NXT [Carletta J. (2005)] is a set of libraries in JAVA that provides support in developing heavily annotated corpus for end user, whether they are multimodal, textual, monologue and dialogue. NXT provides a powerful integrated query language for query based search and retrieval, building graphical user interfaces with command line tools for data analysis.

The IMS Open Corpus Workbench (CWB) [Evert and Team (2010)] formerly known as IMS Corpus Workbench is a set of tools for managing and querying large text corpora with linguistic annotation. Currently, the CWB contains two components:

1. Corpus Query Processor (CQP): This is an interactive command-line program supporting the regular expression, match conditions for all annotation levels, and analysis of collected data. A tool **xkwic**, developed by Oliver Christ at IMS Stuttgart as a part of IMS corpus toolkit. It provide a graphical user interface features in X-windows to support the Corpus Query Processor (CQP).
2. Corpus Encoding Process (CEP): This is deigned for supporting and indexing of the corpus or extracting the frequency information from a CWB corpus, it is including the CWB/Perl packages for simplify and to optimise the corpus encoding procedure.

WordSmith [Scott (2001)] tools is a software to find out the patterns and behaviour of words in a English text. This is used worldwide by the linguist working in the linguistic domain for the various purposes in field such as literature, law, medicine, history, politics and sociology. It can handle most of the frequently used language including English, Chinese, Japanese etc.

It provides the three main programs as:

1. Concordance: Finding all the instances of a word or phrase to see a word in its context.

2. **Keywords:** To find out the salient words for identifying keywords in a text.
3. **WordList:** To generate lists of all words from the texts in alphabetical order with frequency order.

Besides these main functionality, it also provides the functionality of Character Profiler, CharGrams, Corpus Corruption (anomalous in texts), Files Utilities (managing and viewing files), Minimal pairs (for identifies similar words), Text converter (convert text in various format) and Aligner (for translation purpose).

SARA [G. and L. (1998)] is a clientserver SGML based application designed and implemented for the representation and to handle dynamic mapping on services and resources. SARA [G. and L. (1998)] was developed for the access of BNC corpus, but with the help of additional packages, the TEI encoded corpus can be accessed with SARA.

In 1995, Mac Whinney designed a programming tool CLAN [Hillsdale (1995)] for processing the CHILDES database (a multi-language child conversation database). CLAN provide a transcription of child language data, it designed the coding format program for the sharing of transcript data and analysis of data.

The ICE corpus utility program [MEYER (2002)] (ICECUP) is next generation program designed to perform various experimentation on corpus likes corpus exploration, parsed corpora such as ICE-GB. The ICECUP provides a simplicity to conduct hypothesis test together and the various experiments results can be evaluated simultaneously. The salient features of ICECUP are:

CorpusMap: It provides an overview of the corpus, the information of the text is illustrated in hierarchical way, where user can view and extract frequency data as a table of statistics.

Fuzzy Tree Fragments (FTF): The FTF are intuitive way to perform the grammatical queries on the corpus. FTF generates a fragment grammatical tree very similar to the syntactic tree to represent the results of search across the corpus. FTFs tree contain the nodes, words, links and edges to specify the relation among the results. Each node contain the category of words, types of elements, word class while the edges specify the relation between the nodes and words.

3.4.1.3 Prevail for Annotation and Statistical Analysis of Data

The XML based system for corpora development is implemented in the JAVA language. The main goal behind the development of CLARK [Kiril Simov (2003)] is to reduce the human interventions in generating the language resources.

XML technology is the best suitable techniques with document storing, management and querying, because the XML is popular and easy to understand. In multilingual processing task, the core of CLARK is the Unicode based encoding. To generate a hierarchy of XML in DTD style, CLARK applied a mechanism on the Unicode texts. The software implemented an XPath language for the navigation in the document, XPath also help in construct of the input word into grammar XML categorization of the recognised words. The CLARK is mostly used in the corpora mark-up, dictionary compilation for human users and corpora investigation [Kiril Simov (2003)].

A General Architecture for Text Engineering (GATE) [Cunningham et al. (1997)] is a modular system for developing the language processing components. It is comprised of developers, web-application/interface, JAVA library and an architecture of process for the creation of many language development components.

The UCS [Evert (2005)] toolkit is a collection of libraries and scripts for the statistical analysis of co-occurrence data. The UCS toolkit provides the functionality of the measurements or evaluation of the individual or words pairs together in terms of frequency, ranked, manipulating in various way, annotated with association scores, sorted the words in a collocation task.

AntConc [Anthony (2004)] is a freeware software for multipurpose corpus analysis, supporting the execution on multi-platform. AntConc having a powerful concordancer, word clusters for the word and keyword frequency generators. It was first time released in 2002 with the simple KWIC (Key words in context) features at the Osaka University Graduate School of Engineering, the first experiments were conducted on a scientific and technical writing course with contribution of 700 students. The coding was done in PERL for both GUI and interface, so is very easy to port on the Linux/Unix environment.

3.5 Summary

In this chapter, we have given a brief introduction about the software tools needed for the corpus creation to its exploitation.

We addressed the features and important functionality of software in each group. The main focus is given to the tools having corpus annotation and retrieval facility. Here, we address the annotation requirement with various types of annotation procedures adopted with different encoding techniques.

The main goal of all linguists is to develop a tool that would be capable of finding required information from the large database in appropriate way, the classification of the need and use of desired tools can be separated based on the features list and outcomes required with the proposed corpus.

We have given more advantage to our framework by using platform independent interface because most of the tools are platform dependent. Utilization of the web interface for corpus development across all major category, make it more significant for the end-user in terms of learning and adopting the software.

A corpus linguistic approach will be discussed in next chapter, where we will describe the strategies we made for the compilation of our work.

Chapter 4

Compilation of Devanagari Script Corpus

A piece corpus has to be compiled before it can be used. Compilation involves applying a number of tools that process the corpus data so that the complete sketch engine functionality is generated. In this chapter, we detail the compilation process of our unconstrained Devanagari script Hindi corpus.

4.1 Introduction

Presently it is practically improbable to carry out a linguistic study without resorting to a corpus. Compilation of corpus is the process of collecting samples of language in accordance with predefined precedent and putting them together for concordance queries [Krajka (2009)]. The compilation of a corpus is difficult and time consuming. As stated by [Farooqui (2016)], “If data is gathered in an opportunistic way without proper control and documentation, the resulting corpus will be unlikely to be of much use”.

Aggregating of a corpus is very important chore, that requires a specialist for maintaining the growth of design and development. So the design and development of corpus is a immense provocation. That requires a particular enthrallment in respect of experimentation, financial support, durability, easy adoptable etc. In this chapter we detail the corpus compilation by using some criteria related with corpus linguistic to make it more significant like, the need of Devanagari script corpus, genre or category of the data, quantity of data, way of finding data and

collection of data. Beside the above criterion, another significant requirement is equal statistical distribution of information, with proper file structure and keeping in mind to duplicate but at same time also avoid repetition.

Later in this chapter, we explain the motivation to choose the Devanagari Hindi language as source and characteristics of the opted language. We also focus on the constraints applied for the selection of texts, writing constraints, to make it more representative for wide domains and balancing among the criteria. For the transparency perspective of linguistic domain research, a quantitative overview of the corpus is also presented.

India is a multilingual Nation in the world having the population of 1.2 billion according to census report released on 31 March 2011. It is home to the Indo-Aryan and Dravidian language families two of the worlds largest languages. The Austro and Burman language families are also spoken in India. The India map shows three major regions according to majority of the languages. Most of India except the south and north east region uses Indo-Aryan as native language as shown in Figure 4.1.

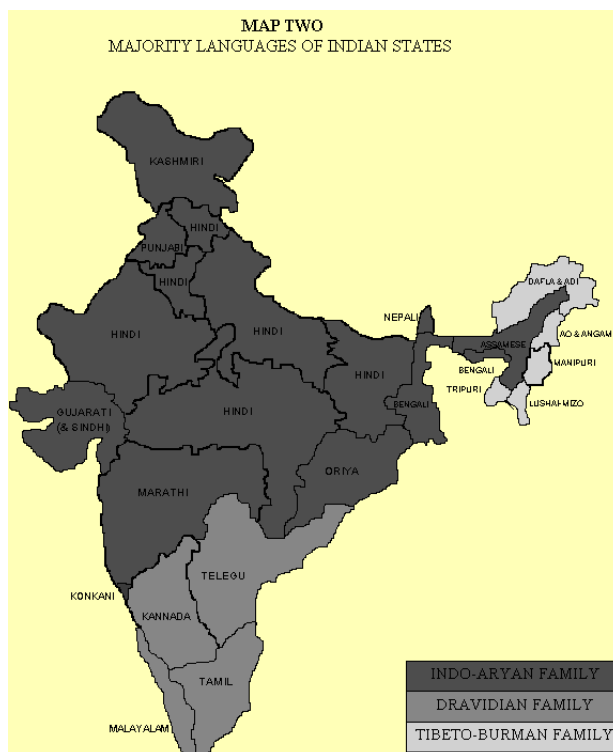


Figure 4.1: The India map show major three regions according to majority of the languages.

4.2 History of Devanagari Hindi

India has 122 major languages and 2371 dialects [Lata (2011)]. Constitutionally 23 languages are recognized (scheduled) including English as associated official language out of which Hindi is one of them. Article 346 of the Indian Constitution recognizes Hindi in Devanagari script as the official language of central government of India. The 23 official languages (and English) are written using 13 different scripts [Bworld (2015)]. Figure 4.2, shows the official language with their script used in different region of India.

Table 4.1: Official language with total speakers and their script.

Sl. No.	Language	Speaker	Script	Writer
1	Assamese	16.8	Bangla	211.5
2	Bengali	181		
3	Manipuri	13.7		
4	Bodo	0.5	Devanagari	328.23
5	Hindi	182		
6	Konkani	7.6		
7	Maithili	34.7		
8	Marathi	68.1		
9	Nepali	13.9		
10	Sanskrit	0.03		
11	Sindhi	21.4		
12	Santhali	6.2	Roman	334.2
13	English	328		
14	Dogri	3.8	Dogri	3.8
15	Gujarati	46.5	Gujarati	46.5
16	Kannada	3.63	Kannada	3.63
17	Kashmiri	5.6	Kashmiri	5.6
18	Malayalam	35.9	Malayalam	35.9
19	Oriya	31.7	Oriya	31.7
20	Punjabi	1.05	Punjabi	1.05
21	Tamil	65.7	Tamil	65.7
22	Telugu	69.8	Telugu	69.8
23	Urdu	60.6	Urdu	60.6

Table 4.1 shows different official languages of India and different scripts used to write these languages with their population distribution in millions. As shown in Table 4.1, Hindi language is the most used Indic script in India.

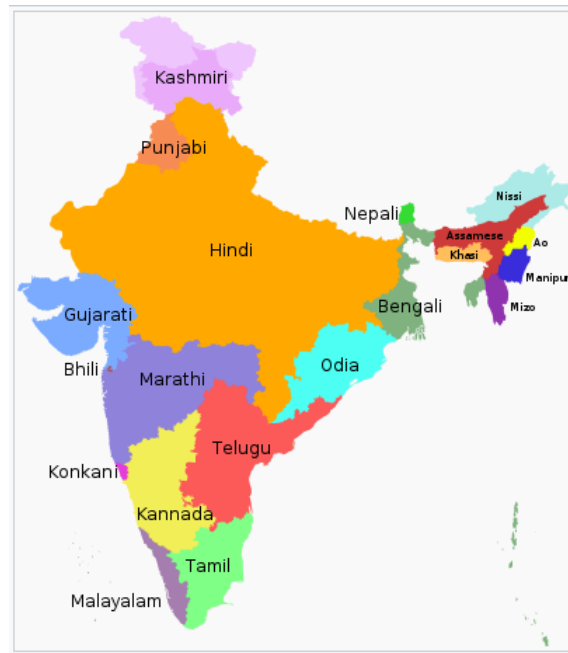


Figure 4.2: A map showing the regions of the Hindi language.

4.3 Motivation Behind Selection of Devanagari Script in Hindi

In the 2001 census, 422 million people in India reported Hindi to be their native language. In accordance with 2011 census, 53.6 % of the Indian population reported that they speak Hindi as either their first or second language, in which 41% of them have declared it as their mother tongue. Hindi (or Hindustani) is the native language of most people living in Delhi, Uttar Pradesh, Chhattisgarh, Uttarakhand, Chandigarh, Himachal Pradesh, Jharkhand, Bihar, Haryana, Madhya Pradesh and Rajasthan.

Census data of India motivated us to work on Devanagari Hindi from Patrika news paper as shown in the Figure 4.3. According to the data, through the country Hindi speaker can't speak another language. So it is necessary to promote first official language of the nation for proper implementation and smoothing working of government schemes at root level.

Linguistic research over Indic Corpus started in 1981. Presently, the availability of the Hindi handwritten corpus is poor. Hindi is the most frequently used language in India and fourth most frequently used language in the world, but due to poor resources the language did not get much attention as compared to other script.

In comparison to other languages Hindi has very poor resources available in digital

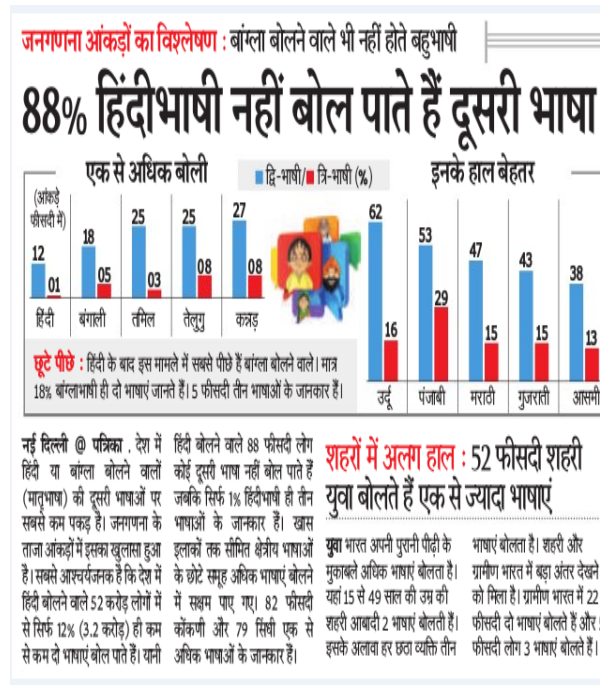


Figure 4.3: Census data of India motivated us to work on Devanagari Hindi from Patrika news paper.

data format. The majority of the data are available only in imprinted format. Most of the online data processing activity either in public or private business or government department are still done manually.

Thus a sundry approach is developed to build Devanagari script Hindi corpus for optical character recognition and demographic data collection. To train a system and fetch data automatically demographic part of data set could be used, that will be beneficial and easier for existing manual data-processing task involved in the field of data collection such as input forms like Voting Card, AADHAR, Census data, Driving licence, Passport, Ration Card, Indian Railway Reservation etc. By this, participation of Hindi language community in understanding and taking benefit of the government schemes will increase.

For the scope and suitability of data set, we designed and developed an approach for data collection, mark-up, digital transcription, linguistic annotation, an XML meta data information, and POS & Chunking.

A benchmarked Devanagari Hindi data set can be of use for linguistic researchers by providing solution for handwritten document. That is an abundant source for digital documentation and preserving of ancient Hindi historical manuscripts of Devanagari Hindi in domain of literature, architecture, art, music and medicine by converting them into digital form and fetch in public domain.

क	ख	ग	घ	ङ	च	छ	ज	झ	ञ
ट	ठ	ड	ढ	ण	त	थ	द	ध	न
प	फ	ब	भ	म	य	र	ल	व	श
ष	स	ह	क्ष	त्र	ळ				

Figure 4.4: A set of Hindi handwritten consonants for Devanagari script in Hindi.

Therefore, an adequate HCR (Handwritten Character Recognition) system is required. To train such systems we require a large database. Thus, availability of a conventional data set for Devanagari Hindi handwritten text recognition is very essential.

In the view, we present an approach for design and development of Devanagari script Hindi corpus, that includes handwritten text sentences having syntactic variations along with demographic information. The corpus is designed in such a way that it structurally marks-up the text page, line and words using coordinates of corresponding segmented images. Besides this structural mark-up, linguistic annotation is also performed. Our framework provides functionality in which handwritten image and transcription information of that image are displayed on the same view port. In ground-truthing and benchmarking of data set, XML file is generated that includes all the meta-data information of demographic and handwritten document.

4.4 Features of Devanagari Script in Hindi

Devanagari script is imitative of ancient Brahmi script which is the mother of almost all Indian scripts. Devanagari is the script in which many languages are written, like the most popular language Hindi, Sanskrit, Konkani, Nepali, Marathi and Sindhi.

The alphabet set of Devanagari script in Hindi has 36 consonants and 12 vowels. The Devanagari script in Hindi consonants are shown in Figure 4.4. The vowels are shown in Figure 4.5. Devanagari Hindi handwritten digits are shown in Figure 4.6. Besides the consonants and vowels it has modifiers also called matras which when combined with vowels form compound characters. The modifiers can be placed on the left or right and above or bottom of a vowel. There are twelve modifiers as shown in Figure 4.5. Devanagari script characters are written in a cursive style

अ	आ	इ	ई	उ	ऊ
ए	ऐ	ओ	औ	अं	अः
व	वि	वी	वु	वुं	वुं
वै	वै	वै	वै	वै	वै

Figure 4.5: A sample of Devanagari Hindi handwritten vowels and modifiers (Matras).

1	१
2	२
3	३
4	४
5	५
6	६
7	७
8	८
9	९
10	१०

Figure 4.6: Devanagari Hindi handwritten digits.

and characters are associated with header lines and usually written on lined paper. The characters are sling from a horizontal line called the header stroke and there are no lower and upper cases like in English characters. Devanagari characters use about two thirds of the space between the lines. In general, the first stroke, or strokes, in a character are written from the left to the right and are then followed by any down strokes and finally the head stroke is added.

The problem is further aggravated by some of the Devanagari character (Hindi) is not used often like ञ, ड, ङ, etc. Some of the characters are written in more than one way like झ.

4.5 Collection and Distribution of Data

The design and development of a Devanagari Hindi corpus starts with the raw data collection, structural annotation (ground-truthing) and ends with proper POS tagging, chunking and labeling of the collected text in the database. In our approach, we used sentence based method rather than collecting a list of isolated characters, digits and words.

Most of the data are collected from the internet, historical-ancient documents, news channels, and text-books. In the poor resource scenario, for Devanagari Hindi Unicode text in some categories like architecture and biographic the printed documents are entered into Unicode text. For maximum variations in the words, sampling period has been chosen since 1910 among seven different categories.

4.5.1 Genre of Texts

The categorization of texts is the task of assigning predefined categories to free-text documents. In paradigmatic situation for peculiar language the corpus consists of a variety of text samples. In our work, the corpus data is a collection of 7 different categories with 19 sub-categories to acquiring the maximum assortment in word collection and to make the corpus more cogent.

There is no definitive criteria for balanced corpus. The criteria we have preferred for balanced corpus is category of texts selection and time extent of data collection. The only benefit of balanced corpus is that texts are selected in such a way that searching becomes more proficient compared to unbalanced corpus. The additional facilities provided by the balanced corpus are classification, filtering and statistical analysis of text for research requirement. That is based on name, age, gender, educational qualification, geographical region and genre of text.

The categories and their respective sub-categories with the symbolic notation of are as follows:

1. News-N
 - (a) International-IN
 - (b) National-NN
 - (c) Sports-SN

2. Science & Technology-S
 - (a) Engineering-ES
 - (b) Medical-MS
 - (c) Physics-PS
 - (d) Chemistry-CS
3. History-H
 - (a) Indian History-IH
 - (b) World History-WH
4. Literature-L
 - (a) Poetry/Religion-PL
 - (b) Shyari/Gazals-SL
 - (c) Biography-BL
5. Politics-P
 - (a) World-WP
 - (b) Central-CP
 - (c) State-SP
6. Architecture-A
 - (a) Rural Architecture-RA
 - (b) Urban Architecture-UA
7. Economy/Business-E
 - (a) Agriculture-AE
 - (b) Industry-IE

4.5.2 Design of Handwritten Form

The layout of the form has been designed in a distinct way to collect a large amount of eloquent information. To provide corpus in the multi-disciplinary research areas of NLP such as: indemnification of writer, verification signature, printed and handwritten text segmentation, OCR algorithms evaluation etc.

To cater to maximum linguistic variations the writers are chosen from different profession, educational qualifications and age groups. The forms filling were carried out at geographically distant locations such as railway stations, shopping malls, hospitals and bus stand etc, as the handwriting of a person sometimes gets affected by the mood, situation and surroundings.

The layout of handwritten form is separated into four parts with a horizontal line for convenience in a segmentation of machine printed text followed by handwritten text and demographic information of the writer. The design of the A4 size form is split into four parts as shown in Figure 4.7, each part is separated from each other with a horizontal line and organized as follow:

The form is divided into four sections by horizontal lines:

- Header:** Contains demographic information: NAME: Vinay, Khandelwal; Profession: student; Age Group: < 20; Gender: Male; Region: Rural; Education: PG.
- Printed Text:** A block of printed text in Hindi, likely a notice or instruction.
- Handwritten Text:** A block of handwritten text in Hindi, replicating the printed text.
- Footer:** Contains the address: Ramgarh, Puchawa, District: Ghazipur, Rajasthan, and a signature.

Figure 4.7: A sample filled in form.

Part 1: The first part of the form (Header) is to collect demographic information of writers which will be an aid for training a system for automatic data entry. In the demographic information, we collected the information of writer as name, age, gender, education, address, rural, urban and date of form filling.

Part 2: After header there is a printed text block, in which text are printed into 3 to 4 lines or 70 to 80 words.

Part 3: Third part of the form is left blank where the writers are asked to write text in own handwriting as shown in Figure 4.8. This part of the form is left blank where the writers replicate the printed text in their own handwriting.

The form is structured as follows:

- Header:** Fields for Name, Profession, Qualification, Age Group, Region, Gender, and Marital Status.
- Printed text Block:** A short paragraph in Hindi script.
- Block for Handwriting:** A large empty space for handwritten text.
- Footer:** Fields for Name and Signature.

Figure 4.8: A sample blank in form.

Part 4: This part of the form consist of the title for a language in the corpus and a Unique Identification (UID). For example, an Hindi language, National news, first form will have an UID as (HIN-N-NN-001). Beside this, address and signature of writers are also to be entered in this part. The UID of the respective form is automatically updated or generated once a language and category/subcategory is selected.

The filled forms were scanned at the resolution of 300 dpi at grey-level using Canon flatbed scanner. Each form was completely scanned, including the printed text, handwritten text and demographic information and its corresponding transcript text of the scanned image is stored in Unicode UTF-8 text files.

4.6 Statistics of the Database

The database consists 1650 handwritten text forms, filled by 1650 writers from different age groups and with different educational qualifications. Text pages are written by both males and females, 75% of the writers are males and 25% are females. Information about the name, age and address were collected on each page. The 75% of 1650 writers were younger than 26 years and 25% were graduate

students. Each writer was asked to write forms in unconstrained environment in their natural handwriting with different multicolor ink and gel pens.

4.6.1 Category Wise Distribution of Text Page

To capture the maximum variance in data collection the domain of data collection is divided into 7 categories and 19 sub-categories. The classification of data collection among 7 category, make it significant by covering large Hindi vocabulary and make it available for the particular areas of interest in corpus linguistic likes science, history, literature, medical, education, economy and architecture. The statistics of the data collection according to category is shows in Figure 4.9. The data follows Gaussian distribution with mean (300 forms) about news category.

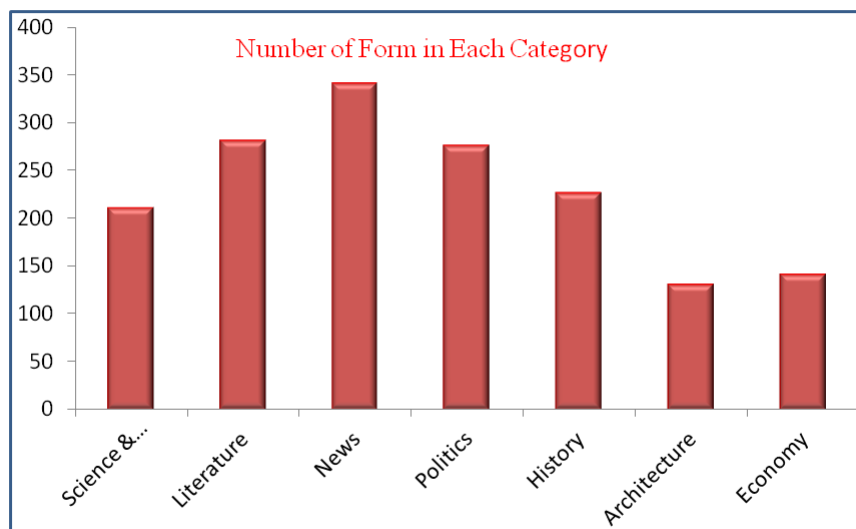


Figure 4.9: Domain wise distribution of handwritten data collection forms.

4.6.2 Geographical Region Wise Selection of Writers

To bring the out the maximum handwriting style variations among the text pages, we selected different geographical region for data collection. This is also meaningful for the OCR research community to provides a complex source for the experimentation and benchmarking. The geographical region wise statistics of the writer

selection is shown in Figure 4.10. Again the data follows Gaussian distribution with mean (290 forms) of Rajasthan state.

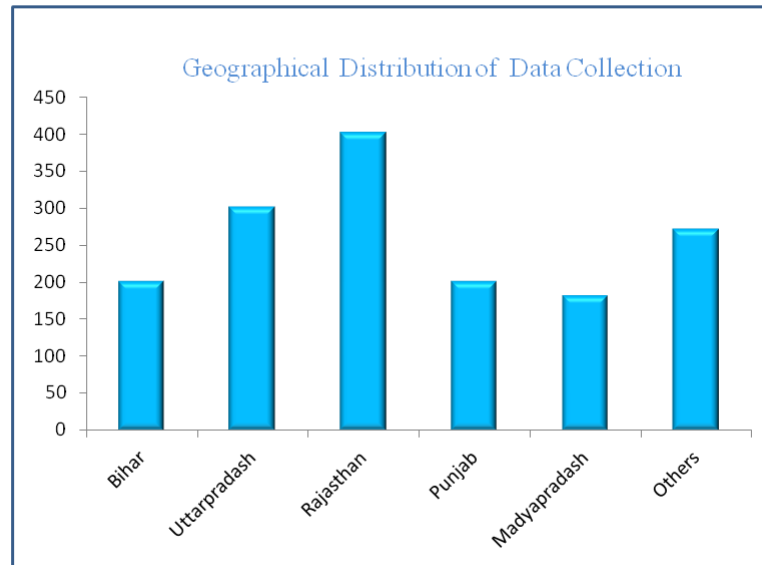


Figure 4.10: Geographical regions wise selection of writers.

4.6.3 Text Distribution Among Category

The database contains 7425 Devanagari Hindi handwritten text-lines and 123,750 words. On average, each filled handwritten text form comprises of 4 to 5 text lines and 70 to 80 text words. In addition of this the database contains 5775 Hindi printed text lines.

The domain wise distribution of lines and words in the database is shown in Figure 4.11. The corpus follows Gaussian distribution for statistical distribution of lines and words. For all three features around 65% of the data is captured under 1-standard deviation, around 93% of the data is captured under 2-standard deviation and 100% of the data is captured under 3-standard deviation. These are also follow the Bell curve.

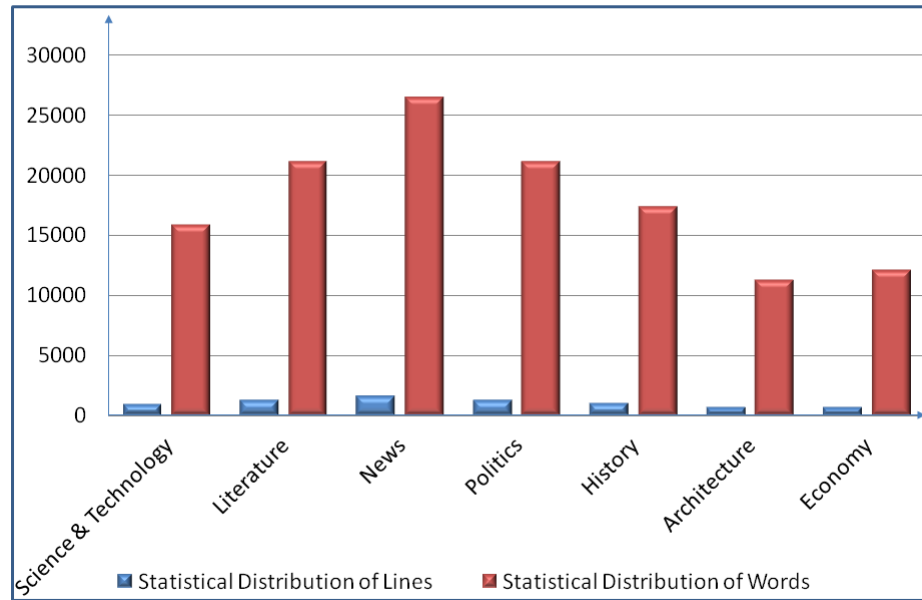


Figure 4.11: Domain wise data distribution of lines and words.

4.6.4 Demographic Statistics

The corpus also contains the demographic information in term of personal and educational information of writers such as name, age, gender, qualification and geographical location. This information makes it more convenient by including various name, city name and numeric data to train the automatic data entry system with actual paradigm. The demographic distribution statistics of the our corpus are shown in Table 4.2.

Table 4.2: Statistics of the demographic distribution of the corpus.

Disparate Demographic information	Writer names	City names	Date formats	Signatures
Total Number	1369	427	332	1650

The educational qualification of most of the writers is an Undergraduate degree, followed by a High school, Secondary school, Ph.D. or pursuing and other. In addition to these, we have also collected information about the writers birth place, city of current residence, as such information may impact the user’s knowledge scope and opinions due to the cultural transfer.

4.7 Summary

We have developed a Devanagari script Hindi corpus for both handwritten and Unicode texts as described in this chapter. A robust approach for data collection and annotation is accounted for in the design and development of corpus in an organized way using the specific framework.

The complexity and challenges faced by Devanagari script Hindi for research work on Hindi handwritten or optical character recognition are detailed here.

The compilation of Devanagari script handwritten text image corpus for Hindi is described in this chapter. Besides this a brief summary about the Hindi language and the reason for the poor resources development in Hindi is provided. The constraint pertained on the corpus to make it more user emblematic are also detailed here.

Chapter 5

Corpus Framework for Encoding and Annotation

In this chapter, we have elucidated the methodology and framework for encoding that has been implemented to annotate the Devanagari Hindi handwritten text image corpus. The features of structural annotation for the implementation of annotation schemes to produce the Ground-Truth (GT) corpus as a resource with experimental results are also represented.

5.1 Introduction

For implementation, we have developed a framework of Devanagari handwritten text corpus to annotate handwritten text documents (offline) in a organized and systematic manner. That performs annotation in minimal time spam besides the data validation. The proposed corpus also contribute some extra linguistics functionality such as ground-truthing of the handwritten image at text-form, line and word levels.

We have explored the backhand data configuration and auto-indexing approach using *MySQL*, that provides consistency among the annotated information, making the corpus efficient in terms of information retrieval and also helpful for the systematic arrangement of information. The user interface and the component architecture, functionality and usability, and comparative analysis of the software available in this field with our work are detailed in this chapter.

5.2 Database System

The NLP is a tract of AI (Artificial Intelligence) and Linguistics, dedicated for making computers understand the statements or words written in human languages. It came into existence to ease the users work and to satisfy the wish to communicate with the computer in natural language. It caters those users who do not have enough time to learn new languages or get perfection in it and have become one of the predominant paradigms in the corpus computational linguistic.

To store the huge amount of data effectively, various software and database strategies are used. According to “Oxford English Dictionary”, a database is “a structured collection of data held in computer storage; especially one that incorporates software to make it accessible in a variety of ways. According to [Elmasri and Navathe (2000)] database is also defined as “a large collection of related data”, where data are facts and statistics collected together for reference or analysis.

A database management system (DBMS) stores data in such a way that it becomes easier to retrieve, manipulate, and generate information. It concedes a user to define, construct and manipulate the database for its own choice and purposes with defined parameters. There are many commercially available database such as: IBM DB2, Microsoft Access, Oracle etc. and freeware such as: SAP, MySQL, DB1 with embedded JavaDB.

5.2.1 Characteristics and Benefits

The data stored in the database describes the structure, definition, constraints and meta-information of the data. A speculative model of the database allows multiple users to access the data without accessing physical storage with different views. The main benefits are to control the redundancy in a database because all the manipulation in data is modelled according to scheme and the access of the database can be controlled based on the users needs, because data is independent of application programs.

5.2.2 The Relational Model of Database

The database information must be demonstrated in a specific manner. There are several data models been proposed, some of the well-known models among these

are the object-based data model, relational data model and network data model etc. The relational model for database management is a technique to manage data using a structure and language consistent. It is the most generally utilized model. It is a simple and elegant model with a mathematical basis.

In Relational Model (RM) the relation resembles a table but the field order of the table are not fixed and operations on the relations follow the operations of the mathematical set theory. The retrievals and updates are two main relations operations based on the functionalities. The retrievals operation are, selection, projections, renames, and joins, while updates operation are inserts, deletions, and modifications of the database.

Here the database consists of a set of tables, where each table includes a set of column and rows called attributes and tuples respectively. A minimal subset of attributes is chosen from the attributes to form an unambiguous identification of each tuple, known as the primary key. The consecutive relation between two tables in RM, foreign key is also used.

SQL (Structured Query Language) is the main benefits of the relational database, it is based on the relational algebra but comparatively not as much complex. SQL can be used independently or with embedded to other languages like PHP, C, COBOL, PASCAL etc.

5.2.3 Our Corpus Database

In augmenting demand of the encoding in XML, the relational database systems are rarely used in corpus linguistics. In our work, we have stored the structural and linguistic annotation information, and original transcript texts of 1650 handwritten text forms using the MySQL database. The main ambition of using this database is to store and develop a corpus query tool. It is beneficial and is a significant way to store the large annotated corpora.

Here, we have chosen the MySQL database management systems because of the MySQL fast access speed and free availability, that supports most of the ANSI features in comparison to other relational databases. It is very fast because data are stored in a B-tree indexing structure. For fast queries in a dynamic situation B-tree structure is designed. That takes $\log_2 N$ to traversing and updating a node. So in large amounts of data B-tree is a very efficient data structures.

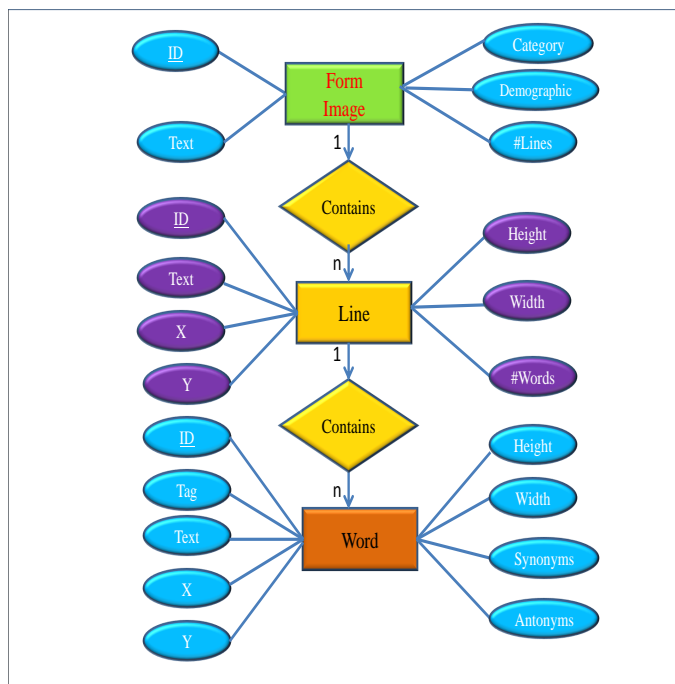


Figure 5.1: ER model of the corpus database.

5.2.4 Entity-Relationship Model (ER Model)

In designing of database flowing four steps are required as:

1. **Requirement Analysis:** The first step of database design is requirement analysis. This step identifies the use of database, functionality required from the database, type of data which will be processed and the most frequently performed operations. The administrator having all rights of update, delete and access of database. The maximum user access the database only for reading purpose and do not perform any other operations.

According to natural language information: a corpus consists of handwritten text- page transcript text, align transcript text of the lines and words form text-pages. Besides these, every word is associated with a grammatical information such as synonyms, antonyms, tag, semantic information. The database also stores the textual region information for the lines and words.

2. **Conceptual Design:** It includes the entity and their relationships to one another. An entity is described as a set of attributes in the database.

In conceptual design, the database is having entities, namely text form, lines and words, each one of them having separate linguistic attributes. The entity text-form has five attributes like ID, texts, number of lines etc. The entity

line and word have seven and nine attributes respectively, likes ID, texts, semantic tag etc.

Text-form includes one or more lines and lines have one or more words. The text-form is annotated in a structured way, and each line, words have exactly one parent text-form, and the tree ends in exactly one root node. In an entity-relationship diagram as demonstrates in Figure 5.1, rectangles represent the entities and relationships are shown by diamond-shaped. Attributes are connected to the entity (rectangles) or relationship (diamond-shaped) and displayed with ovals shape. Primary key attributes are represented with underlined.

Entities and relationships are associated with lines, and labeled with their cardinality. For example; one-to-one, one-to-many, and many-to-many are the basic types of cardinality. Figure 5.1 demonstrates that the cardinality is divided into two parts.

3. **Logical Design:** After the conceptual design, next step is the logical design. In which conceptual data model are formed into the logical one. The logical design is more conceptual, abstract and look at the logical relationships among the objects.
4. **Physical Design:** In this step, we find the most effective way of storing and retrieving the objects from hard disk of computer. It represents how data should be structured and related in a specific DBMS, so it is important to consider the convention and restriction of the DBMS.

5.3 Our Model for Structural Mark-up and Linguistic Annotation

The implementation method and functionality provided by the Devanagari Script Handwritten Text Corpus (DSHTC) are discussed here. Corpus development, annotation (structural and linguistic annotation) and, statistical and linguistic analysis has been divided into three phases as shown in Figure 5.2 and is described as follows:

1. **Raw Corpus Creation:** The first phase of corpus development are associated with the collection and distribution of the raw texts, where texts are

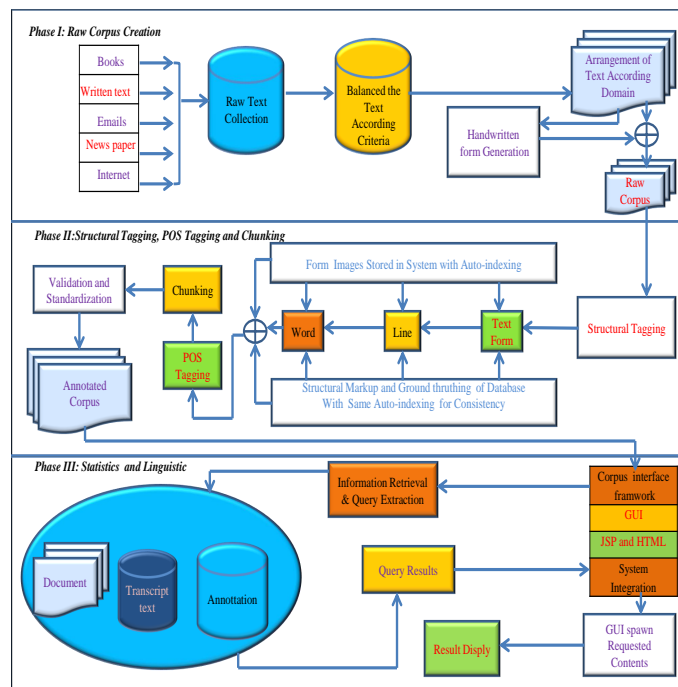


Figure 5.2: The process for corpus building, compilation, structural & linguistic annotation and statistical analysis.

selected randomly from various sources like books, internet, news papers, letters, emails etc, and then the text is chosen and arranged in a spruce way according to categories of text to generate the handwritten text-forms.

- 2. Structural tagging, POS tagging and Chunking:** After flourishing generation of the handwritten text-forms, the second phase describes the structural mark-up, POS tagging and chunking of the compiled handwritten information in electronic format. In which raw collection of text are mark-up in an organized manner using our framework DSHTC with defined set of rules. POS tagging and chunking are detailed in chapter 6 and 7 separately.
- 3. Statistics and Linguistic:** Third and last phase is associated with the verification of information representation and the query based approaches followed by retrieval of the information. The corpus data will be evaluated on standard statistical metrics to validate the text collection. This is described in details in chapter.

The raw text collected in first phase is categorized according to genre and forms are generated automatically. The description of phase second the annotation (structural and linguistic) and amalgamation of information likes images, texts, tagging are detailed in subsequent sections of this chapter.

5.3.1 Structural Mark-up and Corpus Annotation

A Ground-truthing of database with structural and linguistic annotation is an ample amelioration for handwritten text recognition approaches. An annotated corpus is a prerequisite of a wide area of computational linguistic research and provides full support for expert system.

In the structural annotation, besides the text form annotation, DSHTC contribute some further linguistics functionalities such as segmented lines and words aligned transcription. Apart from annotation of handwritten text form, the corpus also stores demographic information of writer of text forms in Unicode format such as: writers name, education qualification, gender, age, and address about the writer of particular form. The annotation of handwritten text forms is performed by standard encoding UTF-8 Unicode. So that to build the corpus in harmony with other Unicode based tools, and to make it platform independent.

5.3.2 Structural Annotation and Auto-indexing

The corpus creation and navigation facilities are provided by structural mapping. Through which all the stored information of handwritten forms, segmented lines and words can be used efficiently. It also contribute further abutment for insertion, alteration and searching of data etc. A process of the corpus designing and its annotation are shown in in Figure 5.3.

The complete structure of each handwritten form unique ID generation is described as follows:

1. A name of file is the sequence of the language (2 bits), -, category (3 bits) and subcategory (3 bits)- xxxxxxxx (8 bit) form no. The indexing configuration is depicted in Figure 5.4.
2. The form UID index is of 16 bits: So the maximum number of forms (total number of forms) = $2^{16} = 65,536$.
3. There can be a maximum of 8 categories, and 8 sub-categories therefore the database contains 2048 forms in each category and 256 forms in each subcategory.
4. Our structure reserves 2 bits for language to additional expand.

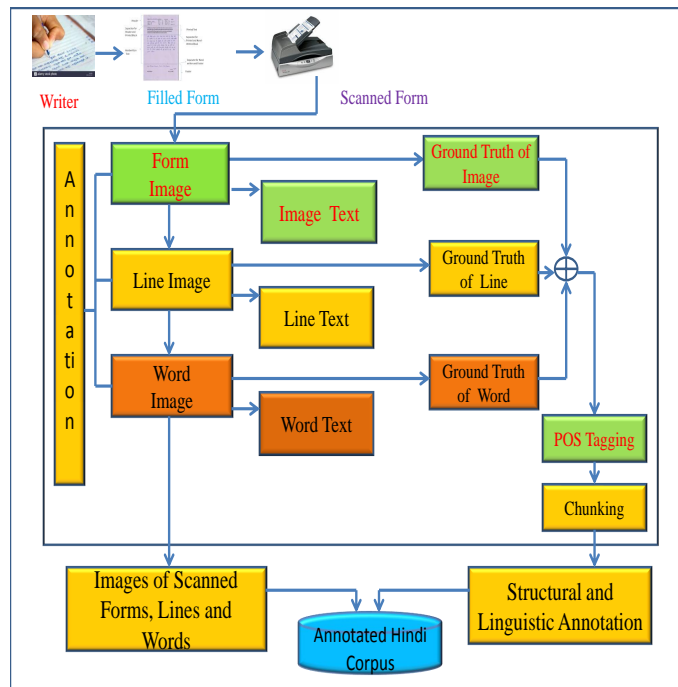


Figure 5.3: A process of the corpus development and its annotation.

Automatically UID genesis of a handwritten text form, to achieve the automatic consistency checking throughout the corpus are depicted in Figure 5.4.

In the process of insertion of a new text form, we can choose a specific script language, category and subcategory of the handwritten text form. Then ID field is appended accordingly. For example, the UID of category “News” in Devanagari script Hindi will be as shown in Figure 5.6. For a particular form s.no.9, UID will be HIN-N-NN-009.

Auto-indexing is also essential for the UID of segmented lines and words of the handwritten form which is an extension of form UID with a notation -. In accordance with form UID, the line and word UID are also auto-generated. For example, a first form UID in category “News” and subcategory “National” of database is denoted as HIN-N-NN-001. The forms image are stored in “PNG” format, for example in category “News” and subcategory “National” will be denoted as HIN-N-NN-001.PNG.

5.3.3 Validation and Ground-truthing

Our framework contributes the feature of aligning the factual location of handwritten texts in the analogous scanned image, lines and words. The textual region coordinates are indexed in corpus along with the XML file. It is convenient for

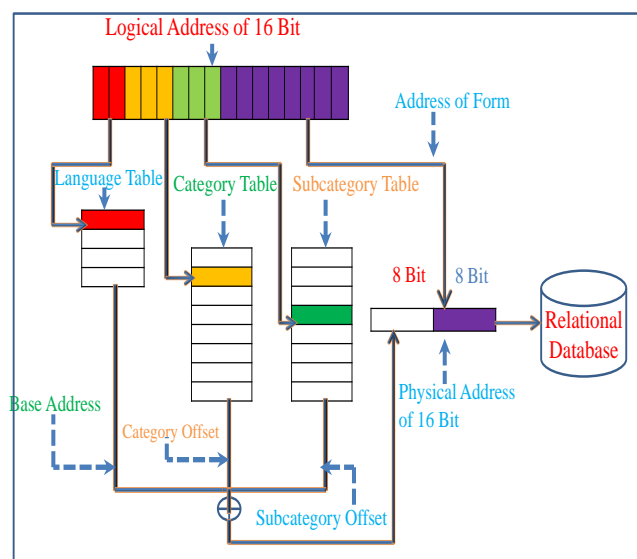


Figure 5.4: Automatically UID genesis of a handwritten text document.

benchmarking of segmentation approaches in handwritten text recognition (HTR). The chosen segmented images of lines and words are stored in a distant folder. The manually recorded Ground-Truth (GT) transcription text of images, lines and words are focused towards the corresponding fields in the data set. The layered procedure of different levels in annotation and *GT* of handwritten text-form, lines and words are shown in Figure 5.5.

The selected textual region are highlighted by displaying a bounding box for better visibility. This is also helpful to perceive the path of image components with a mapping between window screen and view port. During the time when the cursor points at UID (Unique Id) of lines and words etc., a rectangular bounding box show up on image of line and word in the view port as shown in Figure 5.6.

Therefore, the corpus is designed adopting this framework where all information stored in database and all the images of text form, segmented lines and words are stored independently with their respective UID in PNG file format.

To retain the integrity of any database framework validation checks are very important. It is also beneficial for verification that the system operates on accurate and valuable data. With the help of auto indexing and cross-indexing routines applying validation and data normalization rules are equipped in our corpus. The various types of data validations applied to avoid the invalid data entries like form level validation and search criteria validation.

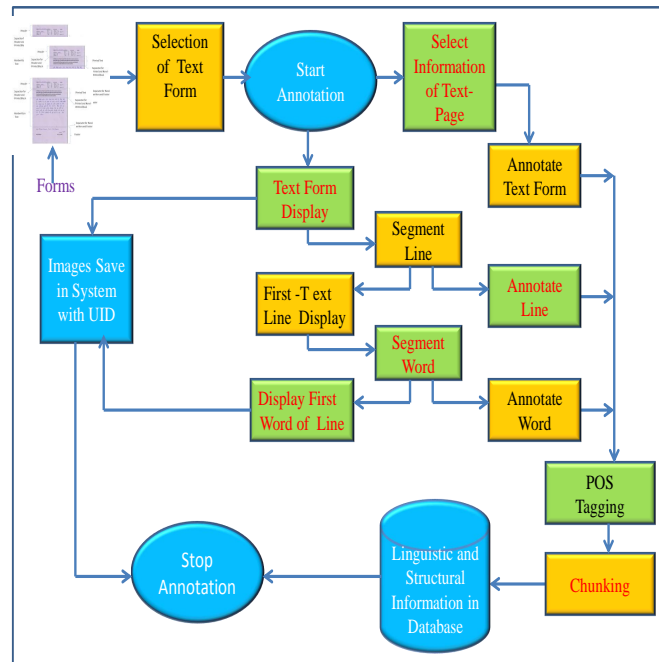


Figure 5.5: The hierarchical iterative process of different levels XML ground-truth generation to annotate the handwritten text-form.

HAND WRITTEN RESOURCES

Category: News SubCategory: International File: HIN_N_IN_0000

MAMAL YACHHVAH JAMHIN ... Qualification: <30 ... Age Group: 30-40 ... Region: Rural

Profession: STIRSH.MT ... Male ... Female ...

सुखी जाते ही सा... अनेक वर्षों में... अनेक वर्षों में... अनेक वर्षों में...

हमें अपने सपने साकार के लिए... अनेक वर्षों में... अनेक वर्षों में... अनेक वर्षों में...

एक... अनेक वर्षों में... अनेक वर्षों में... अनेक वर्षों में...

Language : Hindi

Category : News

Sub-Category : International

File-ID : HIN_N_IN_0000

Text : सुर्धक अपने इस हज़ यात्रा के दौरान कश्मीर के असावा फिलीस्तिन,

Version : 1.0.0

Date Created : 2016-05-02

Date Updated : 2016-05-02

Creator : Hardik

Name : Hardik

Profession : Developer

Gender : Male

Qualification : <30th

Age Group : 20-40

Region : Rural

Address : Ahmedabad

Source : New

No of lines : 2

HIN_N_IN_0000_00 : सुर्धक अपने इस हज़ यात्रा के दौरान कश्मीर के असावा फिलीस्तिन,

HIN_N_IN_0000_01 : अमरीका द्वारा ओसामा बिन लादेन के परचरण की मांग आदि

Figure 5.6: A sample of handwritten text form structural mark-up.

Figure 5.7: Screen shot of form level validation.

Figure 5.8: Screen shot of line level validation.

The validation at the form level, line level etc. are shown in Figure 5.7 and Figure 5.8. It checks the data entry and ensures like the date should not be characters, number of line cannot be empty, GT information is taken automatically and cannot be altered.

5.3.4 Coordinates Information based Mapping

For superior visualization of handwritten text and respective transcript text, a mapping has been done. Mapping of framework is done at different levels: Image, Line and Word.

Figure 5.9: Screen shot to check the null value validation.

We annotate 1,650 text form of our Devanagari script handwritten text corpus with annotation (DSHTC). The details of experimental step is as follows:

5.3.4.1 Mark-up of Text-form

The information associated with each text form is as follow:

1. Unique Identification Number (UID).
2. Transcribed text in the Unicode.
3. Number of handwritten lines.
4. Name of Writers.
5. Rural/Urban.
6. Education.
7. Age/Gender.
8. Address.
9. Source of Information.

5.3.4.2 Mark-up of Line

After selection of a specific line from the side panel, the line is automatically cropped from image and displayed along with the line image and transcription information.

1. UID of line: An auto-indexing number, for example, if image ID is HIN-N-NN-001 than first line ID would be HIN-N-NN-001-01 means first line of image.
2. Transcribed text of line in the Unicode.
3. Pixel information of textual region.
4. Upper X-axis coordinate.
5. Lower X-axis coordinate.
6. Upper Y-axis coordinate.
7. Lower Y-axis coordinate.
8. Width region.
9. Height region.
10. Total words in line.

When cursor move on line UID a rectangle box will show around that line in side panel. This rectangle box will generate in accordance with to our coordinates of X, Y height and weight information which we have generated during the crop of line. As depicted in Figure 5.6, when we select the line, the cropped image of line will display along with the associated information.

5.3.4.3 Mark-up of Word

The word automatically cropped from image and is displayed along with word information such as:

1. UID of word.
2. Textual region pixels information.

3. Transcribed text of word in Unicode format.
4. Synonyms.
5. Antonyms.
6. TAG.

Same process is followed as of line, for the words mark-up. At the time of scrutinizing information user can move back to line for verdict the uses of word in sentence.

5.4 Corpus Encoding Standards

The encoding of the corpus information is a crucial and most important part because the information can be used on different systems regarding different aim. In this regard, encoding should not lose the original information during the retrieval or file transfer.

Several mark-up languages are used in corpus encoding. Standard Generalized Mark-up Language (SGML) was used in many organization, but it suffers information loss due to compatibility problems while transferring text over the web. The SGML has a rigid predefined structure, having some restricted tagsets; it is difficult to handle the complex structure of the document.

5.4.1 Representation of Extensible Mark-up Language (XML)

In the view of all these factors, an XML (Extensible Mark-up Language) was developed by a wide-ranging group of mark-up language experts under the World Wide Web Consortium [(W3C)], which is a software and hardware independent language. It is easy and effective for extracting, storing and transferring data from complex handwritten documents without losing the original contents.

The linguistic and non-linguistic information that we encoded includes as follows:

1. Transcripts text of handwritten text-forms, paragraphs, sentences
2. Textual regions (bounding box) information of each texts

3. Grammatical information of token
4. Meta-textual information such as the characteristics of the document at various level.

The XML is considered with two types of components like low level XML components and high level XML components. In low-level components, there is no bounding on the structure simple tagsets rule are applicable, in a document it contains the features likes paragraphs, sentences and words. The high level components contain features that are used in encoding the documents known as Document Type Definitions (DTD).

In our work high level XML components are used, because it contains the Document Type Definitions (DTD) features that are used in encoding the documents. DTD defines the structure mark-up of the document and its elements as a standard for interchanging data. It is also helpful in validating data encoding.

XML parser can compile and validate the document automatically using the DTD. The hierarchical structure of our DTD specifies that each form should have a corpus title, text-form (annotation information), line (annotation information) and word (mark-up). The documents meeting these features can be considered a text-form of the DSHTC corpus.

Besides this one additional thing we have in the DTD is Text Encoding Initiative (TEI) to provide a standard format for documenting all the necessary information about the text. A TEI is entitled as a header at the top of each DTD documents. XML DTD of the each form includes the header at top that contains the descriptions of the corpus according to the guidelines of TEI.

5.4.2 XML and Corpus Encoding

As we have described that XML based set of rules were used for encoding documents in a format that is both human-readable and machine-readable. XML provides a standard representation which is logically related in a hierarchical way that is a way for document analysis. In corpus ground-truth annotation XML is the most commonly used file format. DSHTC provides the functionality of generating an XML representation based on the data entry provided for each handwritten text form of the database.

Table 5.1: Meta information of an XML formatted file.

Level	Specification	Meta information
1	Demographic information	Writers description
2	Handwritten image	Image unique ID Printed text Date of creation Writer information Number of line Transcription of handwritten text Pixel coordinate of image
3	Segmented lines	Line unique ID Pixel coordinates for bounding box Transcription text Number of words
4	Segmented words	Word unique ID Pixel coordinates for bounding box Transcription text Synonyms, Antonyms

Even though, the DSHTC is the image corpus which includes 1650 scanned images of handwritten full length sentences. In addition to image files, each image is accompanied with a rich XML Meta information file which is encoded with levels of hierarchical meta-information as shown in Table 5.1. The XML schema also encapsulates writers demographic information like name, age, education, address etc.

As a result, the structure generates an XML file for each text-page including the data information of lines and words of respectively pages, based on data entries. For example ground-truth obtained for the handwritten form with UID “HIN-N-NN-001.png” will be “HIN-N-NN-001.xml”.

5.4.3 Procedure of the Corpus Encoding

The CES defines a minimal encoding level that corpora must achieve to be considered standardized in terms of descriptive representation as well as general architecture. The sketch of the overall encoding procedure of our corpus are as follows:

1. The first step is to type text from the source into Microsoft Word and save in Unicode UTF-8.

2. Then counting the words in the text, encode the text with paragraph marker.
3. Subsequently a decision of the elements needed to include in the corpus.
4. The next to paragraph marking, the text was prefixed with the template of the header and suffixed with the closing brackets. A name was given, as national news should be named N01.txt.
5. Fill in the particular information about author and text.
6. Rename the text as .txt to .xml.
7. Open the text in browser for verification of XML file.

5.5 The User Interface

In addition to annotation of raw corpus, the user interface for proposed corpus has been designed in specified manner as corpus structural and linguistic annotation, such as retrieval of information, frequency lists, concordances, exploring the annotated contents.

Interface provide some basic corpus query operations with different query types, instead of learning SQL for the query operations user can directly use the inbuilt interface query operations that have user friendly functionality.

As we have chosen the MySQL database management systems because it provides a fast access speed and is available freely and also supports most of the ANSI features. In comparison to other relational database, MySQL is very fast because of storing the data in a B-tree indexing structure.

The front end implementation part has been done in HTML (Hyper Text Markup Language) and JAVA. MySQL is the most popular database used with JSP, the combination of MySQL and JAVA is platform independent and can be accessed on most of the running platforms like Windows, Linux and Mac operating systems.

An overview of the architecture of our framework is shown in Figure 5.10. It is based on a client and server structure, where the client accesses the server using World-Wide Web. The server is equipped with the Apache web server which includes a module to interpret JSP-script. The JSP script contain SQL-statements to manipulate the MySQL database.

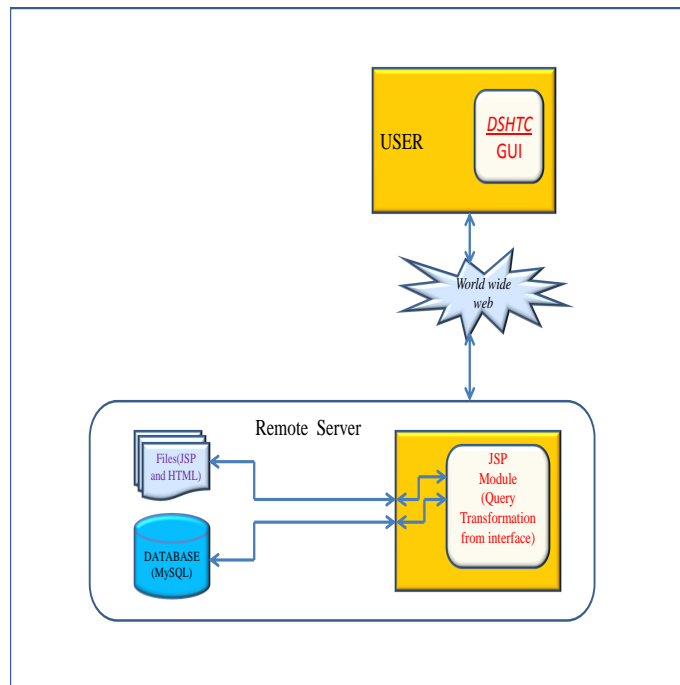
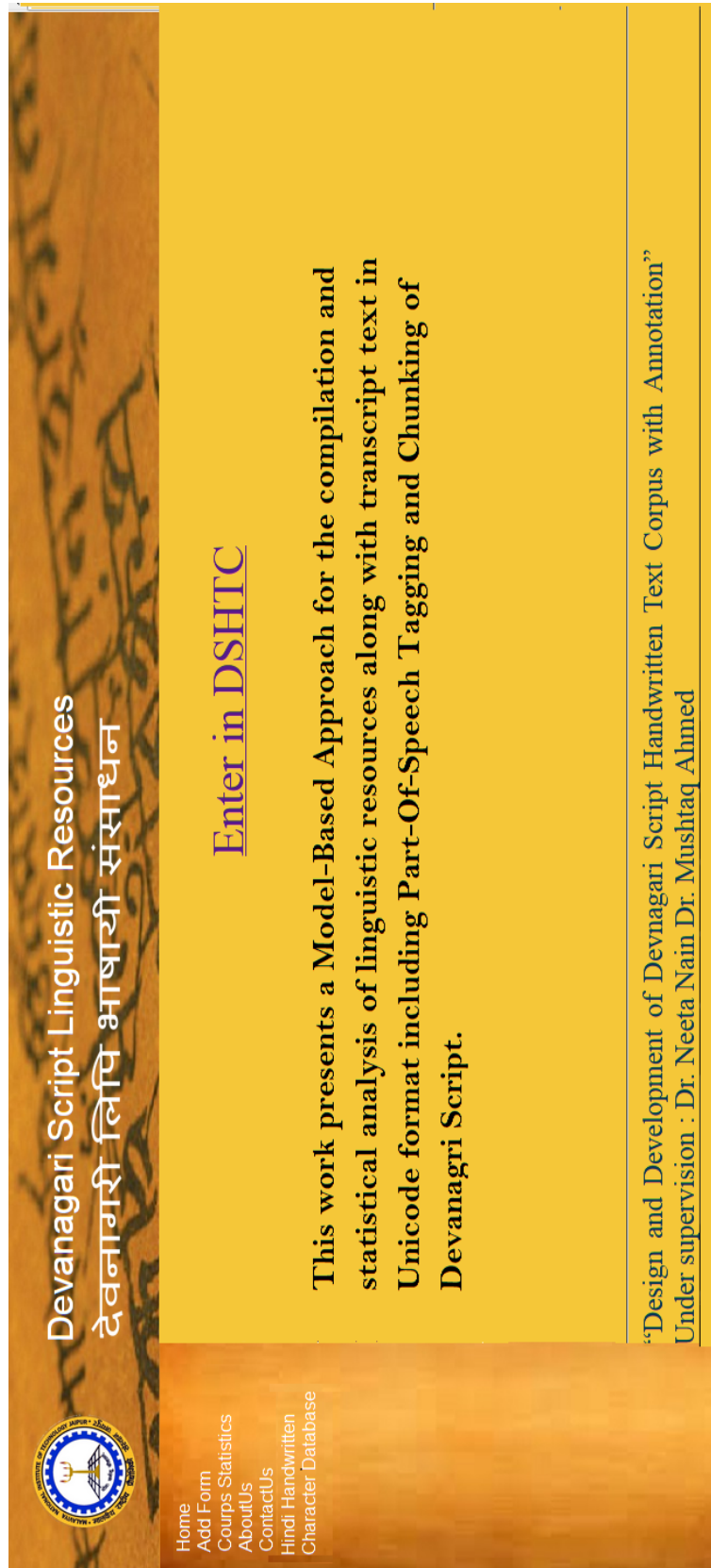


Figure 5.10: Model-based query operations block diagram.

5.5.1 Functions of DSHTC

The interface and output of query functions are perceptible to user. A screenshot of the interface are shown in Figure 5.11.



The image shows a web page with a yellow background and a brown header. The header contains the text "Devanagari Script Linguistic Resources" and "देवनागरी लिपि भाषायी संसाधन" in white. Below the header, there is a navigation menu with links: Home, Add Form, Corpus Statistics, About Us, Contact Us, Hindi Handwritten, and Character Database. The main content area features a blue link "Enter in DSHTC" and a paragraph: "This work presents a Model-Based Approach for the compilation and statistical analysis of linguistic resources along with transcript text in Unicode format including Part-Of-Speech Tagging and Chunking of Devanagari Script." At the bottom, there is a footer with the text: "Design and Development of Devnagari Script Handwritten Text Corpus with Annotation" and "Under supervision : Dr. Neeta Nain Dr. Mushtaq Ahmed".

Devanagari Script Linguistic Resources
देवनागरी लिपि भाषायी संसाधन

[Enter in DSHTC](#)

This work presents a Model-Based Approach for the compilation and statistical analysis of linguistic resources along with transcript text in Unicode format including Part-Of-Speech Tagging and Chunking of Devanagari Script.

Home
Add Form
Corpus Statistics
About Us
Contact Us
Hindi Handwritten
Character Database

“Design and Development of Devnagari Script Handwritten Text Corpus with Annotation”
Under supervision : Dr. Neeta Nain Dr. Mushtaq Ahmed

Figure 5.11: A graphical user interface of Devanagari script handwritten text corpus.

In this corpus, selection of query types includes only the most basic queries. Besides formulating a many of SQL-query statements for one linguistic element of different categories.

The query categories and respective query types executed in our corpus query tool are as follows:

1. Query for word.
 - Query for one word.
 - Query for subsequent word.
2. Query for tag.
3. Query for frequency list.
4. Query for dictionary.
5. Query for the occurrence of word.
6. Corpus-browsing: Query using UID of the text-form, line and word.

5.5.2 Navigation and Information Extraction

The designed Graphical User Interface (GUI) enables the user to select data from the corpus and cater the facility for the developer to annotate the corpus.

User can choose among several options to display the extracted information, or can download the selected portions of the corpus and annotations in their original scanned handwritten format, Unicode or XML format.

The functionality and usability provides for the retrieval of information from the corpus is as follows:

1. A Windows API interface, allowing the mouse-driven queries.
2. Indexing of the handwritten text-images and their transcript text with same UID for the fast online search of corpus data.
3. Extraction of concordance, frequency list, dictionary from the user defined sub-corpus.

4. Display of the word, with uses in a sentence, and count of the total occurrences of word.
5. The inbuilt queries, allow user to perform the flexible query syntax to search the string with terms of Boolean operators, variable intermediate gaps etc.

5.5.3 Corpus Based Education

Corpora are being exploited in educational fields. It is found that teaching basic grammar with a dictionary and real uses of a word is more successful. Corpus has an advantage in this step that, the textual material developed with a corpus are based on real situation example and can be varied.

5.5.3.1 Corpus Based Dictionary

Collins developed a first fully corpus-based dictionary Cobuild [Collins (1995)] for the English language. The corpus was frequently used to build the dictionaries for the languages such as: German [Jones and Tschirner (2005)], Spanish [Davies (2005)], Chinese [Xiao R. and T. (2009)] and French [Lonsdale and Bras (2009)]. An annotated information is associated with each word such as: synonyms, antonyms and grammatical information of the word. The functionality of A to Z, perform a query on the corpus database and display the results of words in alphabetical order with the synonyms and antonyms and the total number of words. The unique ID value of corresponding word could be used to redirect the word in text-page, where it explores the uses of a particular word in the sentence and display the style of writings.

5.5.3.2 Corpus Based E-Learning

From the educational perspective, the corpus is a large collection of documents that covers various topics. One of the fields of application in which corpora are begging to be exploited is as educational one. It is found that teaching basic grammar with dictionary and real uses of word is more successful in comparison to human tutors. Corpus has an advantage in this step that, textual material of the corpus are corpus based on real situation example and can be varied.

The content can be explored on any internet connected devices either via computer or mobile with corpus database stored on the server side. A layer of personalization

of information is worked for filtering and extraction of the desired data from the corpus, and to deliver presentation of requested content according to needs.

DSHTC is a web based structure having potential to use the corpus for teaching and learning purpose and provides facilitates such as:

1. Provide the corpus relevant for teacher and learners.
2. Design corpus-aided activity to retrieval information easily.
3. Enrich in language knowledge and variations.
4. Able to store and explore natural collection of language patterns.
5. Revealing the trends of historical changes in language.

5.6 Comparative Analysis of the Our Corpus Framework

A comparative analysis has been done among the proposed corpus framework with existing ones in two different way as described in below subsections.

5.6.1 Ground-Truth Comparative Analysis

The comparative analysis of our corpus framework with existing ones like Pix Labeller [Saund et al. (2009)], GTLC [Yin et al. (2009)], Truthing Tool [Elliman and Sherkat (2001)], APTI [Slimane et al. (2009)], MAST [T. Kasar and Ramakrishnan (2011)], LabelMe [Russell et al. (2008)] for handwritten text image corpus is depicted in Table 5.2.

The comparatively analysis in Table 5.2 shows the functionality of the existing tools of structural annotation. PixLabler and Truthing Tools provides a way to annotate English language document. APTI and GTLC are available for off line handwritten document annotation in Arabic and Chinese scripts.

APTI has been designed to annotate handwritten image excluding lines and words annotation.

MAST and LabelMe are designed for annotation of camera based images. LabelMe provides the functionality of object recognition in a scene image and MAST can be used for annotation of multi-script scenic images for printed text.

In comparison to the above frameworks DSHTC provides the display of the handwritten Devanagari Hindi text image file and the transcription text of the corresponding image on the same screen in a collaboration context. DSHTC is a simple way for annotation and collection of a huge information for Devanagari script.

The corpus model can be used for different classification criteria as required in multi-disciplinary research like: searching, filtering, statistics analysis on data, and for the study of data distribution in terms of name, gender, education, geographical region domain etc.

Structure	Input Type	Annotations Level	Output	Applications
PixLabeler	English	Image	Text, XML(manual)	Labeling
GTLC	Chinese	Lines, Words, Characters	XML(automatic)	Annotation
APTI	Arabic	Image	XML	Transcription
Truthing tool	English	Image	XML (manual)	Retrieval of text
MAST	Camera Image	Printed text in Image	Unicode XML	Annotation of text
LabelMe	Scene image	Object in image	XML(automatic)	Object detection
IAM	English	Image, Lines, Words	Image XML(manual)	Annotation
DSHTC	Handwritten and Transcript document	Image, Line, Words, POS Tagging, Chunking	Unicode, Auto-generated, XML file(automatic)	Design and annotation (structural & linguistic (POS tagging and Chunking)) corpus, OCR algorithms benchmarking, Unicode transcription, Statistics Analysis of corpus data on various terms, NLP applications

Table 5.2: Comparative analysis of Devanagari script handwritten text corpus (DSHTC).

5.6.2 Functionality Based Comparative Analysis

A Functionality based comparative analysis of the corpus supportive frameworks is useful for different criteria as required in multi-disciplinary research. Table 5.3 shows the functionality based comparative analysis of our framework with existing tools.

Fun.	WC	xKwic	ICECUP	SARA	WS	CB	TACT	BNCWeb	CLAN	DSHTC
Com/Free	Com	Free	Com	Com	Com	Com	Free	Com	Free	Free
OS	D/W	U	W	W	W	OS2	D	U	U/M/D	U/M/D
Text Encoding	Plain	SGML	ICE	SGML TEI	SGML	SGML	SGML	SGML	CHAT TEI	TEI XCES
Indexing	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	No	Yes
Annotation	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
XML	No	No	No	Yes	No	No	No	Yes	No	Yes
Freq. List	Yes	No	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
Comparison of Freq.	No	No	No	No	Yes	No	Yes	No	No	Yes
Statistics Analysis	No	No	No	No	No	No	No	No	No	Yes
Compilation	No	No	No	No	No	No	No	No	No	Yes
GT	No	No	No	No	No	No	No	No	No	Yes
Subcorpus	No	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes
Align Transcription	No	No	No	No	No	No	No	No	No	Yes
POS Tagging, Chunking	No	No	No	No	No	No	No	No	No	Yes

Table 5.3: Comparison of software capabilities for information retrieval.

A brief description of the functionality parameters of the frameworks mentioned in Table 5.3, are as follows:

1. **Operating System:** The considerations of operating systems are important things, because a large number of linguists using the UNIX as their choice and working platform, while some may prefer the Microsoft Windows based features. The choice of operating system is influenced with the graphical interface, need of mouse driven systems for query etc.
2. **Text encoding:** Text encoding is an application to make the corpus in a standard format, which make it feasible for exchange the processing of text information in a significant way. BNC, GATE, SARA, WS and CLAN are using SGML and TEI widely known encoding techniques. In our corpus we have used TEI and XCES encoding standards.
3. **Indexing:** The indexing provide an alphabetically sorted entry of the whole corpus token or word. Before compilation of the entire corpus, user or administrative run an index compilation process to generate the index. The advantage of pre-index is that it improve the speed of retrieval in large size corpus, especially in sequential search. However, the processing of pre-indexing is costly, also time-consuming, it requires additional storage of large index files, and required a necessary skill to complete the task. In our framework, we have applied a structural auto-indexing approach at different levels, to maintain the consistency in stored information.
4. **Annotation:** For annotating a corpus, the software should be intelligent enough especially for the automatic annotation software. However, many annotation tools fail to do this task professionally, and tools are often unable to recognize accurate forms of words. The annotation is an essential part for a corpus development, and various tools adopt different techniques for the annotation. In the annotation process, we have consider both hand-written and Unicode text and provides an align transcription. Besides the grammatical annotation we have also considered the mark-up annotation.
5. **XML:** XML is one of the most needed features for text document processing. BNC, ICE and SARA provides the XML representation for the annotated documents. In our corpus we have provides an automatic XML generated file of annotated information with the DTD schema.
6. **Frequency lists:** Frequency lists are a standard characteristic of any corpus software, it counts the number of words in a corpus. The results of

frequency index are useful for many statistics analysis of the corpus data. GATE, CLAN, WC, ICE etc. provides the frequency of the searched word. WordSmith provides the statistical table according to frequency. BNC provides the frequency results for word and tag. Our model gives the results of frequency for words and tags in a sentence.

7. Statistics Analysis: It is a excellent way for profiling the corpus data, we have applied frequency profiling based approach for the statistical analysis as in chapter 8 of the thesis.
8. Compilation: Development of a corpus with the annotation, BNC and our model provides the facilities of the corpus compilation. We have process both handwritten and Unicode text in the compilation.
9. Ground-truth: Contour level pixel information of the textual region, useful for the benchmarking of handwritten documents.
10. Transcription: In addition of the texts with parent texts in the aligning way (sentence and word), transcription can be done for the handwritten corpus.
11. Sub-corpora: Some tool having ability to represent the some selected portion of the corpus for further analysis about the interested topics. However, this can be done manually by dropping the some files or selecting some portion of texts.
12. Commercial or freeware: Some of the software are freely available on request for the corpus research purpose, while some are sold commercially with flexibility and support. For example SARA, GATE, xKwic etc. are available free for research purpose.

5.7 Summary

In this chapter, a new criteria for data choosing and structural mark-up of corpus are detailed. Here, we have developed a framework that can annotate a large database with prominent effectiveness as compared to other corpus annotation tools.

The ambition of this work is design and development of handwritten document corpora of Devanagari script with structural and linguistic annotation. An exhaustive adumbration of handwritten document image was done in a hierarchy level as: image, line and word.

The developed structure provides an escalating process of annotation in which each level stores the aligned transcript texts of handwritten data and a bounding box coordinate of the textual region.

We present a framework for generating a systematic model for benchmarking of various segmentation and OCR research by providing a ground-truth.

Besides corpus compilation, we have also provided some sort of inbuilt query operation, that could be used for the retrieval of information from the annotated corpus. The query results presented with the above corpus data can be used mathematical models of language.

We have described the comparative analysis of the existing corpora and supportive framework.

Chapter 6

Part-of-Speech Tagging

The automatic annotation of lexical categories is known as Part-of-Speech (POS) tagging. It assigns an appropriate POS tag for each word in a sentence of a natural language. An ample set of linguistically motivated rules or a large annotated corpus are required for the development of an automatic POS tagger.

6.1 Introduction

POS tagging is a important tool for NLP. The annotation of corpus provides it an extra value as we assign linguistic information to a corpus which can be further used in various applications of natural language processing and speech related applications.

POS tagger system generally has three main functions as: breaking of input text in individual sentences, words, and POS tags for input text. It is the process of marking a token or word in a sentence as a particular POS tag or lexical like noun, adverb, pronoun, adjective, number etc. based on its context in sentence, its definition and its morphological information.

There exists different levels of corpus annotation as shown in Figure 6.1. These annotations can be applied separately or can be applied in a sequence [Deepa Modi and Ahmed (2015)]. The annotation levels are described as follow:

1. Morphological level annotation: The study of units of words, their meaning, their internal structure, their forms and how they are used in the sentence.

2. Lexicon level annotation: It is a study of words for a particular language (POS tagging of word).
3. Phrasal levels annotation: Study of phrases of a particular language (Chunk tag of a sentence)
4. Syntactical level annotation: It is a study of syntactical structure of sentence in language.
5. Semantic level annotation: Study of semantic (relation between words) structure of sentence in language.
6. Discourse Analysis: The study of information flow between sentences.

POS tagging is also called grammatical tagging of text. The POS tagging can be divided into two extensive categories: **Closed classes** and **Open classes**. The Closed classes are those that have relatively fixed associations. Examples of POS tagging for Devanagari script text are:

Input Hindi Text:

पेरिस आतंकी हमलों के मास्टरमाइंड के खिलाफ कार्रवाई के दौरान गई महिला फिदायीन के बारे में नए खुलासे हुए हैं और कुछ सामने आए हैं ।

Part-of-Speech Tagging of Text:

पेरिस_NNC आतंकी_NN हमलों_NN के_PREP मास्टरमाइंड_NN के_PREP
 खिलाफ_PREP कार्रवाई_NN के_PREP दौरान_PREP गई_VAUX महिला_NNC
 फिदायीन_NN के_PREP बारे_NN में_PREP नए_JJ खुलासे_NN हुए_VAUX
 हैं_VAUX और_CC कुछ_QF सामने_NLOC आए_VFM हैं_VAUX ।_PUNC.

6.2 Existing Work

The automated POS tagging has been enhanced over the last few decades. Many new concepts have been introduced to improve the efficiency of the tagger and to construct the POS taggers for several languages [Dandapat (2009a)]. In the beginning, people manually engineered the rules for tagging. In recent times several statistical models have been used for the POS tagging for providing portable adaptive taggers. Some of the current research works focus on semi-supervised and unsupervised machine learning models to handle the problem of unavailability of the annotated corpora.

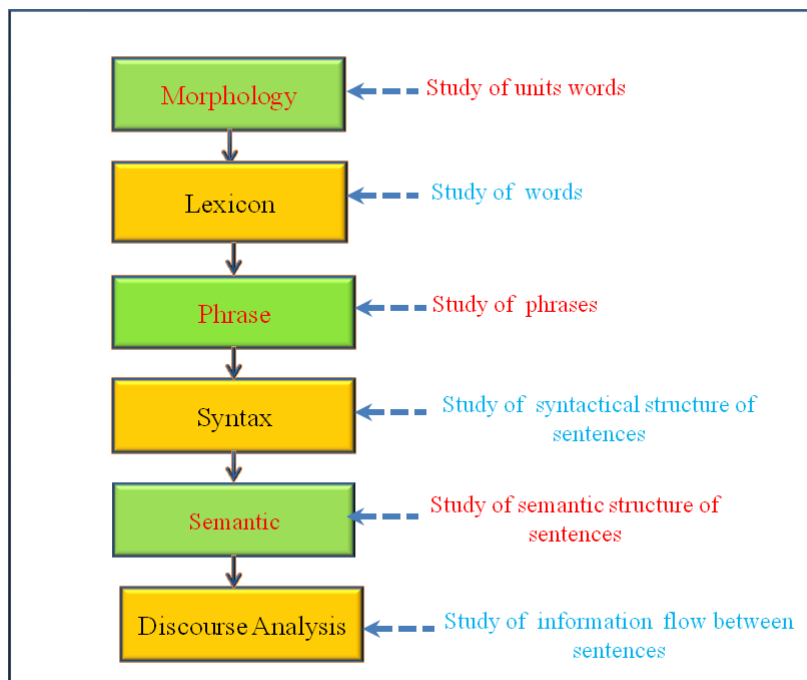


Figure 6.1: Corpus annotation levels.

6.2.1 POS Tagger for Indian Languages

In the recent years there has been a lot of interest in Indian language POS tagging. It was found that poor resources are available for Hindi POS tagging. The comparative study of POS Tagger for Indian languages as shown in Table 6.1 are detailed as follows. The performance result are shown in terms of accuracy and precision.

[Singh et al. (2013)] presented a POS tagger for Marathi language. In this statistical approach was used for POS tagging. For training and testing the model they have calculate frequency and probability of words of given Marathi corpus. To train the system 7000 sentences (1,95,647 words) from tourism domain was used. To measure the performance of systems, a test corpus of 1000 sentences (25744 words) was developed and it achieved 93.82% performance in terms of accuracy.

A hybrid model for Bengali POS tagging are developed by [Dandapat (2009a)]. They have presented a Hidden Markov Model (HMM), Maximum Entropy (ME) and Conditional Random Field (CRF) based models for POS tagging. The training data consists of approximately 40,000 words. The same training corpus is used for all the HMM based tagging schemes. The training data has been annotated using a tag set consisting of 40 grammatical tags. A fixed set of approximately 100,000

words taken from CIIL corpus was used to re-estimate the model parameter during semi-supervised learning. Maximum Entropy based methods can deal with diverse, overlapping features.

[Murthy and Badugu (2013)] presented a new approach of automatic tagging without requiring any machine learning algorithm or training data. They have developed their own Telugu text corpus of about 50 Million words and tested the system on a corpus of 15 Million words.

A POS for Assamese Text is presented by [Saharia et al. (2009)] using Hidden Markov Model (HMM). A tagset of 172 tags in consultation with experts in linguistics and a manually tagged corpus of about 10000 words for training are used. The tagger achieved an accuracy of about 87% for test inputs.

[Pandian and Geetha (2008)] developed a Morpheme based POS tagger for Tamil language. To estimate the contribution factors of the model, a generalized iterative scaling technique is followed. The Model has achieved the overall performance of 95.92%.

[Patil (2018)] developed a Marathi POS tagger using Hidden Markov Model technique. Unigram, bigram and trigram language models are used in the experiment. The HMM model uses Viterbi decoding algorithm for predicting most probable tag sequence for given word sequence for Marathi language. The system is trained using 12,000 sentences and tested using 3000 sentences in Marathi and achieved an accuracy of 86.61% on test data.

A Morpheme based POS tagger for Kannada was created by [Padma and Prathibha (2016)]. The system is tested on some Kannada words taken from EMILLE (Enabling Minority Language Engineering) corpus. To test the performance of the proposed system, four different data sets are created from EMILEE corpus. The EMILEE corpus is a collaborative work of researchers at Lancaster University, United Kingdom and Central Institute of Indian Languages (CIIL), Mysore, India. These data sets contain different types of words like nominal, adjectival, pronominal, verbal inflectional words and derivational words. The performance of the proposed system is above 90%.

[Rishikesh (2018)] developed a POS tagger for the Maithili language, using HMM algorithm. They prepared a training corpus containing 10,000 words and tested the system with more than 1200 known words with 21 tags set. The accuracy of the system is measured to 95.67% for known words.

[Ojha et al. (2015)] presented a POS tagger for Indo-Aryan Languages: Hindi, Odia and Bhojpuri. The system is trained with 90k tokens and 2k tokens test data, collected from ILCI (Indian Language Corpora). The accuracy of the SVM tagger lies between 88% to 93.7% whereas CRF tagger lies between 82% to 86.7%.

[Patel and Gali (2008a)] introduced a POS tagger for Gujarati. In this machine learning part is performed using a Conditional Random Field (CRF) model. This tagger has achieved an accuracy of 92% for Gujarati texts. The system is trained over 10,000 corpus words and tested on 5,000 corpus words.

POS tagger for Bhojpuri language was developed by [Singh et al. (2008)]. A general domain Bhojpuri Corpus was created as part of a research work. That was the first big corpus for Bhojpuri. The tagset for Bhojpuri was designed with 33 tags modeled on BIS standards (annotation scheme for all Indian languages). As per the annotation standards, the training and test data used for testing is in 80 - 20 ratio.

Maximum Entropy (ME) based Bengali POS tagger was developed by [Ekbal et al. (2008)]. The POS tagger has been trained on a corpus of 72,341 words tagged with the 26 POS tags. The POS tagger has demonstrated an accuracy of 88.2% for a test set of 20K words.

[Sharma and Lehal (2011)] improved the accuracy of POS tagger for Punjabi. This HMM approach has been implemented by using Viterby algorithm (VA). The module was tested on the corpus containing 26,479 words and achieved an accuracy of 90.11%.

6.2.2 POS Tagger for Hindi Language

The comparative study of POS Tagger for Hindi language are shown in Table 6.2 are detailed as follows.

[Modi and Nain (2016)] developed a rule-based POS tagger for Hindi language. The system uses a corpus of around 9,000 words and achieved 91.84 % of average precision and 85.45 % of average accuracy.

A rule based POS tagger for Hindi was presented by [Garg et al. (2012)] and yield an accuracy of 87.55%.

A Maximum Entropy Markov Model (MEMM) based POS tagger for Hindi is developed by [Aniket Dalal and Shelke (2006)]. The system is experimented over

Table 6.1: Comparative study of POS tagger for Indian languages.

Method Used	Model Used	No. of Tags	Training Set (No. of Words)	Testing Set (No. of Words)	Accuracy / Precision
Marathi POS [Singh et al. (2013)]	HMM Based Statistical Model	26	1,95,647	25744	93.82%
POS Tagging of Bengali [Dandapat (2009a)]	Hybrid Model	40	40,000	5000	92.37%
POS Tagger for Telugu [Murthy and Badugu (2013)]	Rule Governed	NA	50 Million	15 Million	91.5%
Assamese POS Tagger [Saharia et al. (2009)]	HMM Model	176	10000	NA	87%
Tamil POS Tagger [Pandian and Geetha (2008)]	Morpheme Based Model	35	NA	36,128	95.92%
Marathi POS Tagger [Patil (2018)]	HMM Model	IIITH Tag Set	12,000	3000	86.61%
Kannada POS Tagger [Padma and Prathibha (2016)]	Morpheme Based Language Model	BIS Dravidian	EMILE	NA	90%
Maithili POS tagger [Rishikesh (2018)]	HMM Model	21	10,000	1200	95.67%
POS Tagger of Indo-Aryan [Ojha et al. (2015)]	SVM & CRF Model	BIS	90K	2K	93.7%
POS tagger for Gujarati [Patel and Gali (2008a)]	CRF Model	26	10000	5000	92%
POS Tagger of Bhojpuri [Singh et al. (2008)]	SVM Model	33 BIS	90k	NA	88.6%
Bengali POS Tagger [Ekbal et al. (2008)]	Maximum Entropy Model	26	72,341	15K	88.2%
Punjabi POS Tagger [Sharma and Lehal (2011)]	HMM Model	630	8 Million	26,479	90.11%

corpus of NLP AI-ML 2006 [Aniket Dalal and Shelke (2006)] contest consisting of around 35000 words annotated with 29 different POS tags. The system achieved 89.346% accuracy for POS tagging.

[Shrivastava and Bhattacharyya (2008)] developed a HMM based POS tagger for Hindi, using a naive stemmer and achieved an accuracy of 93.12%. This system does not require any linguistic resource apart from a list of possible suffixes for the language.

[Singh et al. (2006)] produced morphologically rich POS tagger for Hindi using CN2 algorithm with a precision of 93.45%.

[Dalal et al. (2007)] designed feature rich POS tagger for morphologically rich Languages based on maximum entropy and Markov model with the 94.38% precision.

A domain specific CRF based POS tagger for Hindi are presented by [Gupta et al. (2016)]. In this an agriculture based corpus, of 1000 sentences in English language were chosen based on agriculture. The 100 sentences are tested using CRF. The accuracy of the overall system was achieved as 54.16%.

[Mohnot et al. (2014)] developed a Hybrid POS tagger for Hindi Language. The system is evaluated over a corpus of 80,000 words with 7 different standard part of speech tags for Hindi. It has a performance with an average accuracy of 89.9% for POS tagging.

[Mishra and Mishra (2011)] developed a POS tagger for Hindi. If Hindi word is not classified in any category of POS tags then they tagged that word with No Tag.

A Hidden Markov Model based POS tagger for Hindi is presented by [Joshi et al. (2013)]. Where IL POS tagset proposed by [Akshar Bharati (2006)] is used. The system is trained over 15,200 sentences (3,58,288 words) from tourism domain. The system performance is tested on a corpus of 500 sentences and achieved 92.13% accuracy.

A Quantum Neural Network (QNN) based POS tagger for Hindi is designed by [Narayan et al. (2014)]. To analyze the effectiveness of the proposed approach, 2600 sentences of news items having 11500 words from various newspapers have been evaluated.

[Pandian and Geetha (2008)] developed an improved statistical POS tagging using Hidden Markov Models (HMM) for Hindi and Telugu. In this Brants Tri-

Table 6.2: Comparative study of POS tagger for Hindi language.

Method Used	Model Used	No. of Tags	Training Set (No. of Words)	Testing Set (No. of Words)	Accuracy / Precision
Part-of-Speech Tagging of Hindi Corpus [Modi and Nain (2016)]	Rule-based Model	29	9,000	3,189	91.84 %
Rule-Based Hindi Part of Speech Tagger [Garg et al. (2012)]	Rule-based Model	30	18249	26149	87.55%
Hindi POS Tagging and Chunking [Aniket Dalal and Shelke (2006)]	Maximum Entropy Markov Model	29	35000	8750	88.4%
Hindi POS Tagger Using Naive Stemming [Shrivastava and Bhattacharyya (2008)]	Hidden Markov Model	NA	53400	13500	93.12%
Morphological-Constructing a POS Tagger for Hindi [Singh et al. (2006)]	CN2 Algorithm	27	15562	3890	93.45%
Building Feature Rich POS Tagger [Dalal et al. (2007)]	Maximum Entropy Markov Model	27	15,562	3890	94.38%

grams'n'Tags (TnT) HMM based tagger and CRF++, a CRF based tagger are used. The system achieved 94.21% and 91.95% accuracy for Hindi and Telugu respectively.

[Baskaran (1998)] developed a Hidden Markov Mode based POS tagging approach and achieved 76% precision in tagging.

6.2.3 POS Tagger for Other Languages

[Gunasekara et al. (2017)] created a hybrid POS tagging model for Sinhala language using HMM and rule-based approach with overall accuracy of 72%.

Table 6.3: Comparative study of POS Tagger for other Languages.

Method Used	Model Used	No. of Tags	Training Set (No. of Words)	Testing Set (No. of Words)	Accuracy / Precision
POS Tagger for Sinhala [Gunasekara et al. (2017)]	Hybrid Model	UCSC Tagset	75,830	25,087	79.08%
POS Tagger Nepali [Yajnik (2017)]	HMM and VA	41	45,000	15005	95.43%
Arabic POS Tagger [Farrah et al. (2018)]	Hybrid Model	21	NA	NA	92%
Indonesian POS Tagger [Muljono and Supriyanto (2017)]	HMM Model	31	10,000	3000	94.95

A statistical approach based POS tagging for Nepali text are developed by [Yajnik (2017)]. In this system POS tagging for Nepali text used Hidden Markov Model and Viterbi algorithm. The database is generated from NELRALEC [Yajnik (2017)] with 41 tags. The system is trained over database of 45,000 Nepali words with corresponding tags and tested over 15005 randomly collected words. The system achieved accuracy of 95.43%.

[Farrah et al. (2018)] improved POS tagging for Arabic with hybrid approach. The POS tagger presented in this work is based on regrouping rules-based learning aspects as well as machine learning through neural networks with 21 tags shared between different categories.

[Muljono and Supriyanto (2017)] developed POS tagger for Indonesian language. In this Indonesian part-of-speech tagging morphology analyzer and HMM are combined to improve the performance of tagging. The system is trained over 10,000 tokens and tested over 3,000 tokens to achieve better performance in comparison to other methods. The comparative study of POS Tagger for other than Indian and Hindi language are shown in Table 6.3.

6.3 Techniques for POS Tagging

There are various types of techniques for POS tagging and Chunking, all these techniques can be classified in two broad categories, like supervised learning and unsupervised learning. Both learning techniques are further divided in three major types of techniques, i.e., stochastic based technique, rule-based technique and hybrid technique. Supervised stochastic approaches cover various models like Hidden Markov Model, Memory based model, unigram model etc [Antony (2011b)].

Where as for unsupervised stochastic approach there is very less work has been done, because this type of technique only classifies the input text but does not provide tags to input text.

Stochastic based tagging generally uses very large databases of tagged corpus and large pre human made tables to tag new incoming data, where as rule-based technique uses hand-crafted rule patches and does not require pre-saved data.

Although rule-based approaches require deep knowledge of language and are difficult to develop but these techniques are memory efficient and take less time to develop the system [Hasan (2006)]. Figure 6.2 show the classification of these techniques.

Supervised POS tagging and chunking is a machine learning methodology, which uses a pre-tagged, human made corpus as training data. These techniques can be performed using HMM and SVM based taggers [Hasan (2006)]. Unsupervised POS tagging and chunking methods do not use any pre-tagged, human made corpus. These kind of taggers use some advanced computational methods to automatically generate transformation rules for different classes. On the basis of available information and data these type of methods find language dependent rules to calculate probabilities [Hasan (2006)].

6.3.1 Rule Based Approach

The rule-based method is one of oldest methods of POS tagging and chunking. The rule-based POS tagging model applies a set of handwritten rules and uses contextual information to assign POS tags to the words. It uses hand-crafted language rules or transformational rules for tagging corpora. Rule-based taggers depend on a pre-annotated dictionary to get all possible tags for each word in the input sequence. When a word has more than one possible tag, then hand-crafted

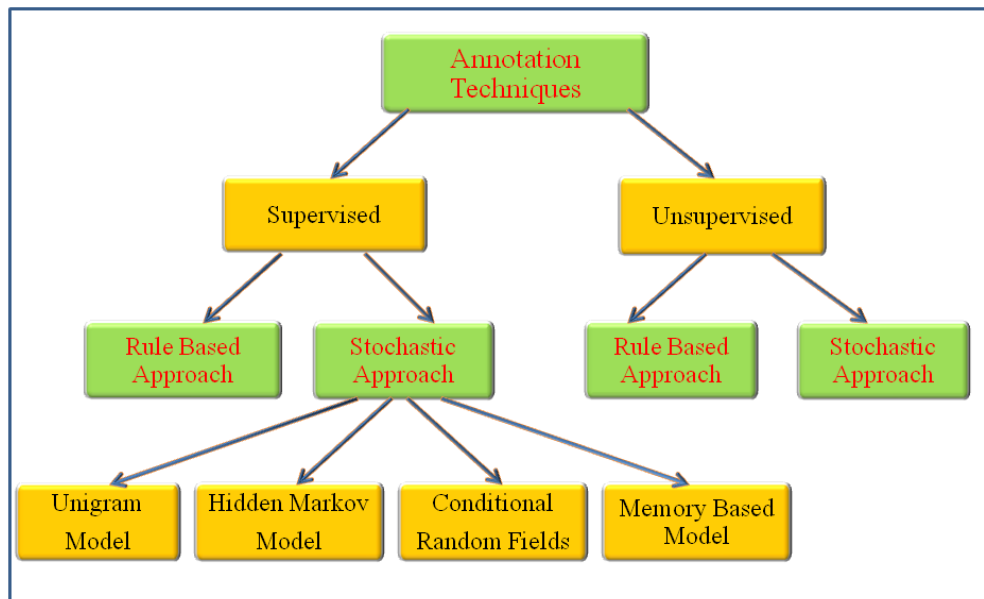


Figure 6.2: Various techniques for POS tagging.

language rules are used to identify the correct, unambiguous tag for the word. For disambiguation linguistic features of the word are used and it also depends on its preceding word, following word and many other aspects features [Garg et al. (2012)].

In these approaches, very less amount of stored information is required. In these types of taggers, improvements are very easy to apply. These methods provide easy portability of system from one language to another. With these advantages, these approaches also have various disadvantages too, as these rules are difficult to construct, typically not very robust and to some extent, these are language dependent.

6.3.2 Stochastic Approach

A stochastic model predicts a set of possible outcomes weighted by their likelihoods, or probabilities [Taylor and Karlin (1998)]. The most popular approaches nowadays use statistical or machine learning techniques. These approaches primarily consist of building a statistical model of the language and using the model to disambiguate a word sequence.

The language models are commonly created from previously annotated data, which encodes the co-occurrence frequency of different linguistic phenomena [Dandapat (2009a)].

The problem with this approach is that it can come up with sequences of tags for sentences that are not acceptable according to the grammar rules of a language [Antony (2011a)].

For a given sentence W in input text, consisting of a sequence of words w_1, w_2, \dots, w_n ; stochastic methods determine the most probable sequence of tags (POS tags and Chunk tags) $T = t_1, t_2, \dots, t_n / C = c_1, c_2, \dots, c_n$. Where t_1, t_2, \dots, t_n are the POS tags and c_1, c_2, \dots, c_n are Chunk tags.

These approaches require a large-sized pre-annotated, human-made corpus and according to this corpus, it finds most probable tag sequence or chunk sequence for input word sequence. These methods require pre-calculate frequencies or statistics of every word existing in the training data. There can be various types of stochastic-based method, some of them are described here.

6.3.2.1 Unigram Model

Unigram taggers are based on a simple statistical approach. In this each token, assign the tag that is most probable for that particular token. For example, it will assign the tag JJ to any occurrence of the word frequent, since frequent is used as an adjective more often than it is used as a verb [Bird (2009)]. A unigram model (n-gram, $n = 1$) used for corpus annotation looks like the combination of several one- state finite automata.

Unigram model (lexical model) states that POS category or chunk category of the current word depends only on its own tag and is independent of its context or in other words tag for a word or phrase does not depend upon its previous tags [Deepa Modi and Ahmed (2015)].

$$P_{Unigram}(t_1 \ t_2 \ t_3) = P(t_1)P(t_2)P(t_3) \quad (6.1)$$

Unigram probability of a word w_i with tag t_i can be calculated using following equation,

$$P(t_i | w_i) = \frac{freq(w_i | t_i)}{freq(w_i)} \quad (6.2)$$

Unigram model maximize the value of $P(t_i | w_i)$ to find most probable tag t_i for word w_i , where $t_i = t_1 \cdots t_n$ and (w_i, t_i) exists in corpus C . These methods use

pre-tagged corpus for tagging.

6.3.2.2 Hidden Markov Model

Hidden Markov Model (HMM) is a statistical technique that can be used to solve classification problems that have an inherent state sequence representation. The HMM model includes an interconnected set of states which are connected by a set of transition probabilities [Dandapat (2009a)].

According to [Zhou and Su (2002a)], HMM is characterized by five elements as:

1. The number of distinct states (N) in a model. That denotes the individual state as $S = S_1, S_2, \dots, S_N$.
2. The number of distinct output symbols (M) in the HMM. The individual symbol are denoted as $V = v_1, v_2, \dots, v_M$.
3. The state transition probabilities $A = \{a_{ij}\}$. The probability a_{ij} , is the probability of moving state i to j in one transition.
4. The observation symbol probability $B = \{b_j(k)\}$. The probability $b_j(k)$ denotes the probability that the k^{th} output symbol will be emitted when the model is in state j.
5. $\pi = \{\pi_i\}$, the initial state distribution, where π_i is the probability that the model will start at state i.

This model takes an input as a word sequence $W = w_1, w_2, \dots, w_n$ and assigns most probable tag sequence $T = t_1, t_2, \dots, t_n$ to the input sequence. HMM finds out the most probable sequences. It considers all the possible sequences of tags and out of all possible sequences, provides the most appropriate tag sequence [Zhou and Su (2002b)].

According to Baye's rule,

$$P(T | W) = P(T) \times P(W | T) \quad (6.3)$$

For maximum probable sequence it becomes,

$$P(T | W) = \operatorname{argmax}_t P(T) \times P(W | T) \quad (6.4)$$

By considering simple assumption that relation between a word and its tag is independent of its context, Word likelihood probabilities $P(w_i | t_i)$ can be defined as,

$$P(w_i | t_i) = \frac{\operatorname{freq}(t_i | w_i)}{\operatorname{freq}(t_i)} \quad (6.5)$$

And Tag transition probability $P(T)$ for bigram model can be defined as,

$$P(T) = p(t_1)p(t_2 | t_1) \cdots p(t_n | t_{n-1}) \quad (6.6)$$

Tag transition probability $P(T)$ for n-gram model,

$$P(T) = p(t_1)p(t_2 | t_1) \cdots p(t_n | t_{n-m}; \cdots t_{n-2}, t_{n-1}) \quad (6.7)$$

6.3.2.3 Maximum Entropy Markov Model

This model allows the computation of $P(t | h)$ for any t from the space of possible outcome T ; for every h from the space of possible histories, H . The computation of $P(t | h)$ in maximum entropy depends on a set of possible features which are helpful to predict the outcome [Dandapat (2009a)].

The disadvantage of this approach is the label bias problem. The probabilities of transition from a particular state must sum to one. Maximum Entropy Markov Model (MEMM) favors those states through which less number of transitions occurs.

The probability distribution that satisfies the above property is the one with the highest entropy [Ekbal et al. (2008)].

It is unique, agrees with the maximum likely-hood distribution, and has the following exponential form:

$$p(t | c) = \frac{1}{Z} \exp\left(\sum_{i=1}^n \lambda_i f_i(c, t)\right) \quad (6.8)$$

where, t is the POS tag, c is the context (or history), $f_j(c; t)$ are the features with associated weight λ_j and $Z(c)$ is a normalization function. The problem of POS tagging can be formally stated as follows. Given a sequence of words $W: w_1 \cdots w_n$, we want to find the corresponding sequence of POS tags $T: t_1 \cdots t_n$, drawn from a set of tags \mathcal{T} , which satisfies:

$$P(t_1 \cdots t_n \mid w_1 \cdots w_n) = \prod_{i=1,2,\dots,n} P(t_i \mid c_i) \text{ where, } h_i \text{ is the context for the word } w_i.$$

6.3.2.4 Conditional Random Fields

A Conditional Random Field (CRF) is a construction of a probabilistic class model to divide and tag a sequence of data. It is a type of discriminative probabilistic model and specifies the probabilities of all possible tag sequences given an observation (word) sequence.

The conditional probability of a tag sequence calculated by CRF model depends on non-independent, arbitrary features of the observation sequence.

The probability of a transition between states can depend not only on the current word but are also on previous and next word of the sequence [Hasan (2006)]. Conditional Random Fields, defined as the undirected graphical models, are frequently used to assign tags to input sequence by calculating the conditional probability of values on output nodes and given input values on input nodes.

According to [Lafferty (2001)] CRF is defined as, 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: $p(Y_v \mid X, Y_w, w \neq v) = p(Y_v \mid X, Y_w, w \sim v)$, where $w \sim v$ means that w and v are neighbors in G .

[Patel and Gali (2008b)] formulated CRFs as follows. Let G be a factor graph over Y . Then $p(y \mid x)$ is a conditional random field if for any fixed x , the distribution $p(y \mid x)$ factorizes according to G . Thus, every conditional distribution $p(y \mid x)$ is a CRF for some, perhaps trivial, factor graph.

If $F = A$ is the set of factors in G , and each factor takes the exponential family form. Then conditional probability of a state tag sequence $Y = y_1, y_2 \cdots y_n$ given

an observation sequence $X = x_1, x_2, \dots, x_n$ is calculated as:

$$P(y | x) = \frac{1}{Z(x)} \left[\exp \sum_{i=1}^n \sum_j \lambda_j f_j(y_{i-1}, y_i, x, i) \right] \quad (6.9)$$

Here $f_j(y_{i-1}, y_i, x, i)$ is feature function and λ_j is weight which is learned via training.

6.3.2.5 Memory Based Model

Memory Based Learning (MBL) techniques are lazy learning techniques. Pre-solved examples are saved in memory and processing is postponed till a new instance or problem appears for solving. In annotation, the MBL approach takes pre-tagged data as input, and produces a lexicon and memory based tag as output.

The performance component is used to match the similarity of the input word with the output of the learning component to produce the actual output of the system [Hasan (2006)].

Stochastic methods have various advantages and disadvantages. In these models all necessary and required statistics can be automatically acquired easily from already trained corpus.

6.3.3 Hybrid Approach

The hybrid approach is one, which may perform better than statistical or rule based approaches. The hybrid approach first uses the probabilistic features of the statistical method and then applies the set of hand coded language rules.

6.4 Our Part-of-Speech Tagger

Our approach is designed using a combination of rule-based and stochastic technique. A complete process of the whole approach is depicted in the Figure 6.3. The input for our part-of-speech tagging system should be in Devanagari Hindi and details of the approach are as:

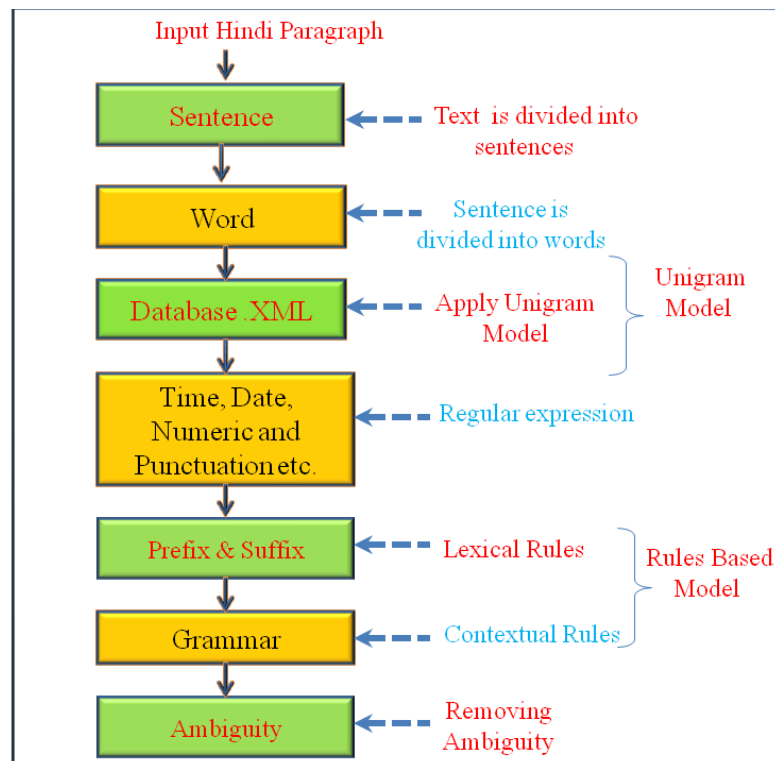


Figure 6.3: The flow graph of our approach.

1. The system accepts Devanagari Hindi text, so the first step is to verify whether input text, is in Devanagari Hindi or not. When it is verified that input text are in Devanagari Hindi, than the system process further.
2. After the text verification, in second step it finds likeness of every word of the input text with the already trained language dependent corpus and tries to find a match. If a match is applicable then respective tag is assigned to input word.
3. In third step, the system pursuit various regular expressions in input text like numbers (786), punctuation marks (?, !,\$), date (05-01-13), time (10:15:09), and special symbols(*, #, \$\$) etc., and assigns the a particular POS tag to that input text. This kind of matching of regular expressions is very good as it increases the precision and accuracy of the system significantly.
4. As the final step, the POS tagging system applies various lexical rules and contextual rules. The lexical rules works on the presumption that the tag for a word rely upon only on current word and not dependent on previous and next words and their tags, based on suffixes and prefixes of Hindi language to assign tags to the remaining unknown words.

6.4.1 System Functionalities

A large system in terms of size and functionality are very difficult to manage and thus is subsystems or modules. The modularity delineates degree to which system components can be alienated. If the depth of the modularity increases, the system becomes more convenient and quite simple to handle. The system has following functionality.

1. Reading and Verification of Hindi Text: The very first module of system are reading and verification of Hindi data. It consists a text area in GUI.
2. Split in Sentences: This module breaks input Hindi data in particular sentences in accordance to delimiter, which can be “Puranviram” or “Prashanvachak chinha”.
3. Tokenize the Words: This module breaks the input Hindi data in to individual words according to delimiter “space. The input will be in Devanagari Hindi sentence text and output will be individual Hindi words.
4. Tag Hindi Data: This module tags each word of input Hindi (Devanagari) data with tags like pronoun, adverb, date, number, verb, time etc. Words which are not tagged using corpus matcher or various rules are tagged as “SYM” tagged which is later tag using rule-based approach.

6.4.2 Unigram Model

In the our POS tagging method the unigram and rules-based approaches are used with 29 tag set as shown in Table 6.4. A manually developed data set of Hindi is used in the system. The system is trained over 11,600 Hindi words with their respective POS tags. The system work in two phases. In first phase it tags known words according to trained data set and in second phase it tags unknown words using rule base approach.

In the first phase, the POS tags for a word is assigned according to unigram model. Each token, is assigned the tag that is most likely for that particular token. A unigram model (n-gram, n=1) is used for corpus annotation looks like the combination of several state finite automata.

Unigram model states that POS tag of the current word depends only on its own tag and is independent of its context or in other words tag for a word or phrase does not depend upon its previous tags [Deepa Modi and Ahmed (2015)].

Unigram probability of a word w_i with tag t_i can be calculated using following equation,

$$P(t_i | w_i) = \frac{\text{freq}(w_i | t_i)}{\text{freq}(w_i)} \quad (6.10)$$

Unigram model maximize the value of $P(t_i | w_i)$ to find most probable tag t_i for word w_i , where $t_i = t_1 \cdots t_n$ and (w_i, t_i) exists in corpus C. These methods use pre-tagged corpus for tagging.

6.4.3 Rules Based Model

We can see from prior work that most of the taggers are developed using statistical methods because these methods are easy to implement and require very less knowledge about the language.

In the second phase of our POS tagger the unknown words are tagged using the rules based on various lexical and contextual features. Which are derived by calculating probabilities of words, their previous words, next words and combination. These described in detail in the following subsection.

6.4.3.1 Rules Based Upon Lexical Features

The lexical feature based rules include rules based on prefixes, suffixes, and on regular expressions. These rules find a particular pattern in the input text. Lexical features are not effected by the context of the current word.

1. Rules Based on Regular Expressions: These rule follow different FSM automata's for matching particular patterns. In these rules patterns are searched in the input text and assigned tags accordingly as shown in the Table 6.5.

A separate FSM automaton exist for each pattern. These rules search patterns like punctuation mark (. : ; |), special symbols (# \$ %), numerical data (930 735 211) etc. in the input text and assign tags accordingly.

Table 6.4: The tag set for Hindi with 29 tags.

S. No.	Tag Description	Annotation Convention
1.	Noun	NN
2.	Proper Noun	NNP
3.	Pronoun	PRP
4.	Verb Finite Main	VFM
5.	Verb Auxiliary	VAUX
6.	Verb Nonfinite Adjectival	VJJ
7.	Verb Nonfinite Adverbial	VRB
8.	Verb Nonfinite Nominal	VNN
9.	Adjective	JJ
10.	Adverb	RB
11.	Noun Location	NLOC
12.	Postposition	PREP
13.	Particle	RP
14.	Conjunction	CC
15.	Question Words	QW
16.	Quantifier	QF
17.	Number Quantifiers	QFNUM
18.	Intensifier	INTF
19.	Negative	NEG
20.	Compound Common Nouns	NNC
21.	Compound Proper Nouns	NNPC
22.	Noun in Kriya Mula	NVB
23.	Adj in Kriya Mula	JVB
24.	Adv in Kriya Mula	RBVB
25.	Interjection Words	UH
26.	Punctuation Marks	PUNC
27.	Time	TIME
28.	Date	DATE
29.	Special	SYM

Table 6.5: Patterns and with their tag.

S.N.	Type of Patterns	Pattern	Tag
1.	Punctuation Marks	., ,, :,	PUNC
2.	Special Symbols	\$, %, #	
3.	Numeric Data	345, 930, 79399	QFNUM
4.	Date Format	06/01/13, 06-01-2013	DATE
5.	Time Format	10:29:59, 10:25:57:59	TIME

Table 6.6: Some examples of rules using prefixes.

S.N.	Prefix	Tag	Example
1.	अधि	Noun	परिपूर्ण, परिवार, परिश्रम
2.	उन	Noun	उनतीस, उनासी, उनचास
3.	अधि	Noun	अधिपति, अधिकार, अधिकरण
4.	अन	Noun	अनजान, अनमोल, अनाचार
5.	अति	Adjective	अतिथय, अतिरेक, अतिसार
6.	आ	Noun	आकार, आजीवन, आजीविका
7.	उप	Noun	उपमान, उपयोग, पकार
8.	भर	Noun	भरमार, भरपेट, भरसक
9.	अनु	Noun	अनुज, अनुक्रम, अनुकरण
10.	अप	Noun	अपमान, अपकर्ष, अपकार

Table 6.7: Some examples of rules using suffixes.

S.N.	Suffix	Tag	Example
1.	वाला	Noun	फलवाला, हिम्मतवाला, घरवाला
2.	ई	Noun	लालची, धनी, क्रोधी
3.	आस	Noun	खटास, मिठास
4.	अक	Noun	सेवक, कृषक, लेखक
5.	आई	Adjective	बुनाई, कमाई, लिखाई
6.	शाली	Noun	शक्तिशाली, वैभवशाली, बलशाली
7.	आका	Noun	धमाका, लडाका, कडाका
8.	आलू/आलु	Noun	दयालु, झगडालू, कृपालु
9.	आवट	Noun	सजावट, लिखावट, मिलावट
10.	आहट	Noun	घबराहट, चिल्लाहट, गुराहट

$$Func(word, tag) = RE_Finder(pattern, word) \quad (6.11)$$

2. Rules Based on Prefixes: There are many words in Hindi which starts with prefixes like **अव**, **अप** etc. These words can be tagged with high probability tag like Noun, Adjective etc.

$$Func(word, tag) = Search_{prefix}(word) \quad (6.12)$$

3. Rules Based on Suffixes: In these rules, the words are also tagged with high probability as in prefixes based rules.

$$Func(word, tag) = Search_{suffix}(word) \quad (6.13)$$

6.4.3.2 Rules Based Upon Contextual Features

These rules are chosen from Hindi grammar and based on various combinations of previous, current and next tag depending upon the words context.

$$Func(word, tag) = Apply_Rule_{combi}(tag_{i-1}, tag_i, tag_{i+1}) \quad (6.14)$$

The rules are as follows:

1. If current tag is postposition then the previous tag will be probably the noun.

Example: सोहन ने पानी में फुल देखा। In this example, “ने” is postposition and “पानी” is noun.

2. If the current tag is adjective then the next tag will be probably the noun.

Example: सोहन खुबसुरत लडका है। In this example, “खुबसुरत” is adjective and “लडका” is noun.

3. If the current tag is the verb like Finite Main, Nonfinite Adjective, Nonfinite Adverbial or Nonfinite Nominal, then the previous tag will be probably the noun. **Example:** वह शहर जा रहा है। In this example, “जा” is verb and “शहर” is noun.

4. If the current tag is pronoun then next tag will be probably the noun.

Example: यह हमारी कार है ।

In this example, “हमारी” is pronoun and “कार” is noun.

5. If the current tag is noun and the next tag is proper noun then the current tag will be probably compound proper noun.

Example: संजिव अगवाल जा रहा है । In this example, “संजिव” is compound proper noun and “अगवाल” is proper noun.

6. If the current tag is auxiliary verb then the previous tag will be probably the finite main verb. In the previous example “रहा” is auxiliary verb and “जा” is main verb.

7. If two consecutive tags are noun then the first tag will be probably compound common noun.

Example: केंद्र सरकार ने बेरोजगारों के लिए काम नहीं किया। In this example, “केंद्र” is compound common noun and “सरकार” is noun.

8. If the current tag is the verb and the previous tag is either noun, adjective or adverb then the previous tag is changed to a noun in kriya mula, adjective



Figure 6.4: Ambiguity removing process in POS tagging approach.

in kriya mulla or adverb in kriya mulla respectively.

Example: हमने फल लाल होते ही तोड़ लिया। In this example, “होते” is verb and “लाल” is adjective in kriya mulla.

6.4.4 Ambiguity Removing

A word can have more than one meaning and according to its meaning in the given context it can have more than one POS tags in different sentences. So the ambiguity removing rules are implemented in the POS tagger. The process is depicted in Figure 6.4. The ambiguity removing process checks the context of the ambiguous words and assigns or re-assigns correct tags to the words accordingly. Example: आम आदमी आम बहुत पसंद करते हैं। In this sentence the word “आम” appears twice but have different POS tag at both places. The ambiguity removing process checks the context of first appearance of “आम”, its next tag is noun so “आम” is tagged as adjective. While for the second appearance of “आम”, whose next tag is not noun so “आम” is tagged as noun.

6.4.5 Experimental Setup of POS Tagging Method

Any POS tagging system generally have three main functionalities like breaking of input text in individual sentences, in words and POS tagging for input text.

Our system provides these functionalities with 100% correctness of splitting input text into sentences and tokenizing. Experiments have been performed to check validity are as follows:

1. Experiment 1: Splitting

This experiment states system functionality of split input Devanagari Hindi text in individual sentences according to delimiter full stop or question mark. Considering the following example,

Input text to the system:

पाकिस्तान के बर्खास्त पूर्व प्रधानमंत्री नवाज शरीफ के मामले में भी यहां चर्चा होगी। पत्रकारों के लिए कल्याण कोष नई दिल्ली। वित्तमंत्री यशवंत सिन्हा ने अपने बजट में पत्रकारों के लिए कल्याण कोष स्थापित करने की घोषणा की है।

Output of the system:

HIN_N_IN_0004_1

पाकिस्तान के बर्खास्त पूर्व प्रधानमंत्री नवाज शरीफ के मामले में भी यहां चर्चा होगी।

HIN_N_IN_0004_2

पत्रकारों के लिए कल्याण कोष नई दिल्ली।

HIN_N_IN_0004_3

वित्तमंत्री यशवंत सिन्हा ने अपने बजट में पत्रकारों के लिए कल्याण कोष स्थापित करने की घोषणा की है।

2. Experiment 2: Tokenization

This experiment states system functionality to “Tokenize” input Hindi text into individual words according to space delimiter. Considering the following example,

Input Text to the system:

पाकिस्तान के बर्खास्त पूर्व प्रधानमंत्री नवाज शरीफ के मामले में भी यहां चर्चा होगी।

Output of the system:

पाकिस्तान, के, बर्खास्त, पूर्व, प्रधानमंत्री, नवाज, शरीफ, के, मामले, में, भी, यहां, चर्चा, होगी, ।

3. Experiment 3: POS Tagging

This experiment shows system functionality of “POS” tagging of input Hindi word sequence. Considering the following example,

Input Text to the system:

पाकिस्तान के बर्खास्त पूर्व प्रधानमंत्री नवाज शरीफ के मामले में भी यहां चर्चा होगी।

Output of the system:

पाकिस्तान_NNP के_PREP बर्खास्त_JJ पूर्व_JJ प्रधानमंत्री_NN नवाज_NNPC शरीफ_NNP के_PREP मामले_NN में_PREP भी_RP यहां_NLOC चर्चा_NVB होगी_VFM ।_PUNC

The complete experimental setup of POS tagging is depicted in Figure 6.5. Sample results of splitting and tagging are shown in Figure 6.6 and Figure 6.7 respectively.

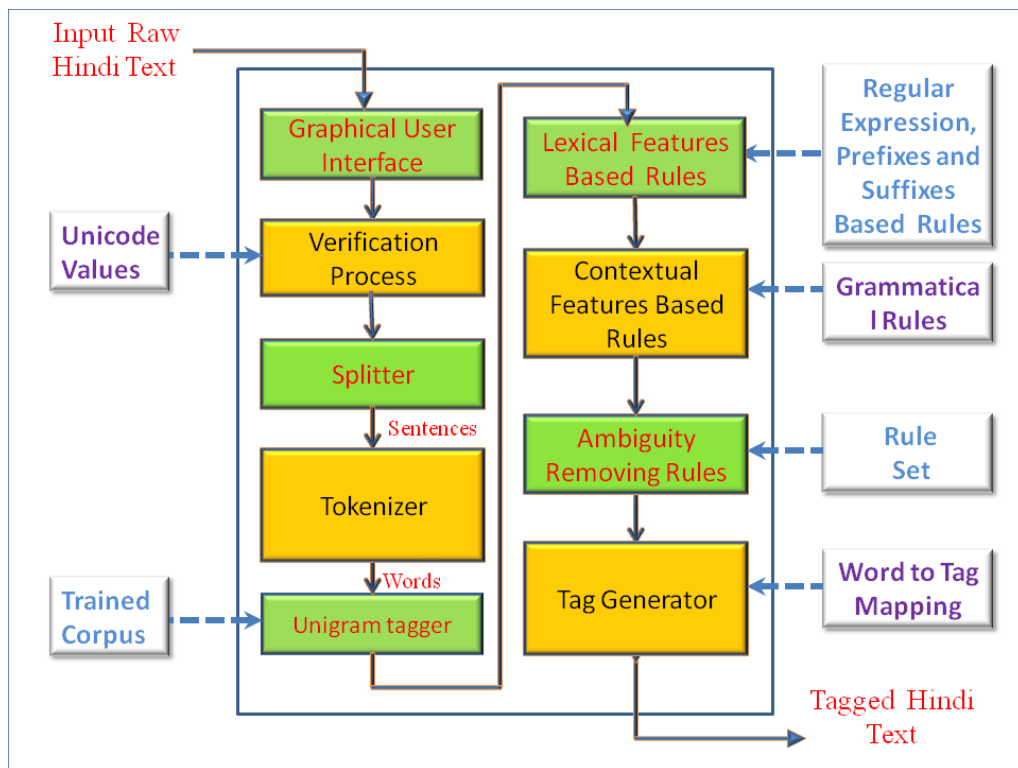


Figure 6.5: Experimental framework of our POS tagging approach.

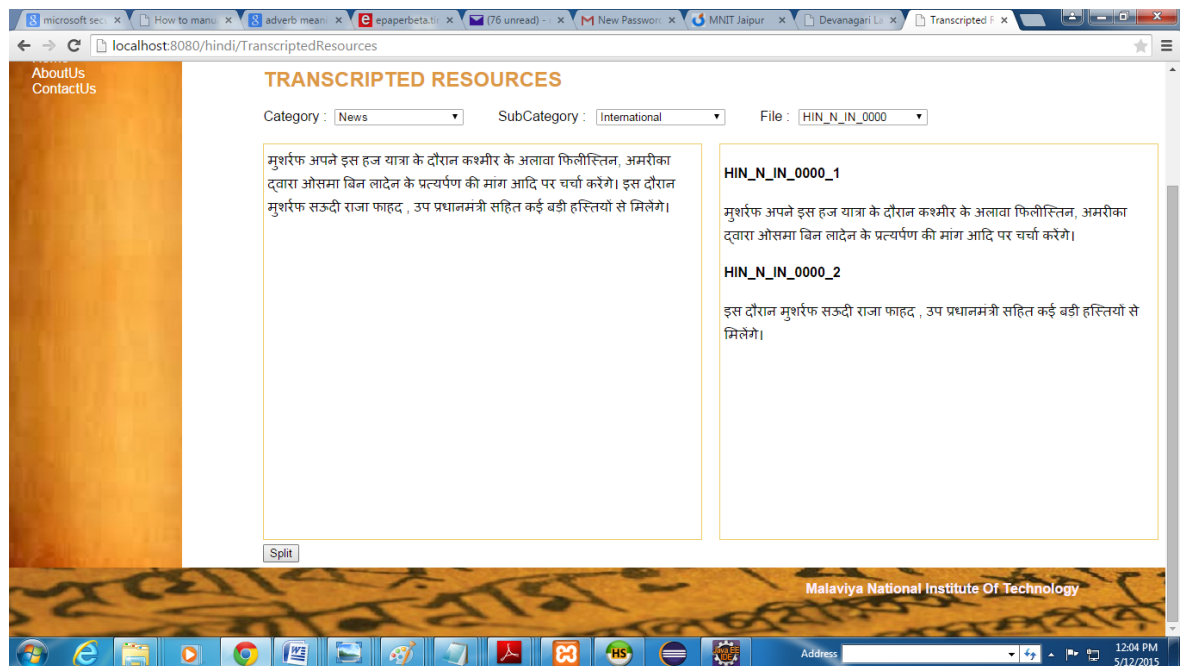


Figure 6.6: Sample result of splitting the input text.



Figure 6.7: Sample result of POS tagging.

6.4.6 Performance Result of POS Tagging Method

Hindi text is collected from various genres like news, science and technology, history, literature, politics etc. and collected from resources like news, online data, stories, books, newspaper, articles etc. The test data are around 24% of the training set.

The system was validated on data set as shown in Figure 6.8, through holdout method of cross-validation. In this method the complete data set is divided into two sets, with 75% as training and 25% as testing. Rules are derived from the

Table 6.8: Performance result of POS tagging approach in terms of P-Precision, A-Accuracy and S-Sensitivity.

Domain	No. of Words	No. of Tagged Words	Correctly Tagged Words	P	A	S
Science & Technology	340	327	311	95.10	91.47	96.17
Literature	500	463	435	94.38	87.4	92.60
News	662	558	538	96.42	81.26	84.29
Politics	516	487	471	96.71	91.27	94.37
History	382	360	341	94.72	89.26	94.24
Architecture	226	168	157	94.64	70.35	74.33
Economy	236	201	187	95.02	80.93	85.16



Figure 6.8: Data set of different domains for testing.

training set, and the upcoming data are tagged according to these rules. Evaluation measures of the system are precision, accuracy and sensitivity as defined by [Nawwaf (1999)].

$$Precision = \frac{\text{No. of Correctly Tagged Words}}{\text{Total No. of Tagged Words}} \quad (6.15)$$

$$Accuracy = \frac{\text{No. of Correctly Tagged Words}}{\text{Total No. of Words}} \quad (6.16)$$

$$Sensitivity = \frac{\text{Total No. of Tagged Words}}{\text{Total No. of Words}} \quad (6.17)$$

The performance of POS tagging is shown in Table 6.8. The proposed system achieved 95.28% average precision, 84.56% average accuracy and 88.73% average sensitivity. The domain wise comparative results of precision, accuracy, and sensitivity are graphically depicted in Figure 6.9.

The comparative analysis of our tagger with state-of-art Hindi part-of-speech taggers such as [Modi and Nain (2016)], [Garg et al. (2012)], [Aniket Dalal and Shelke

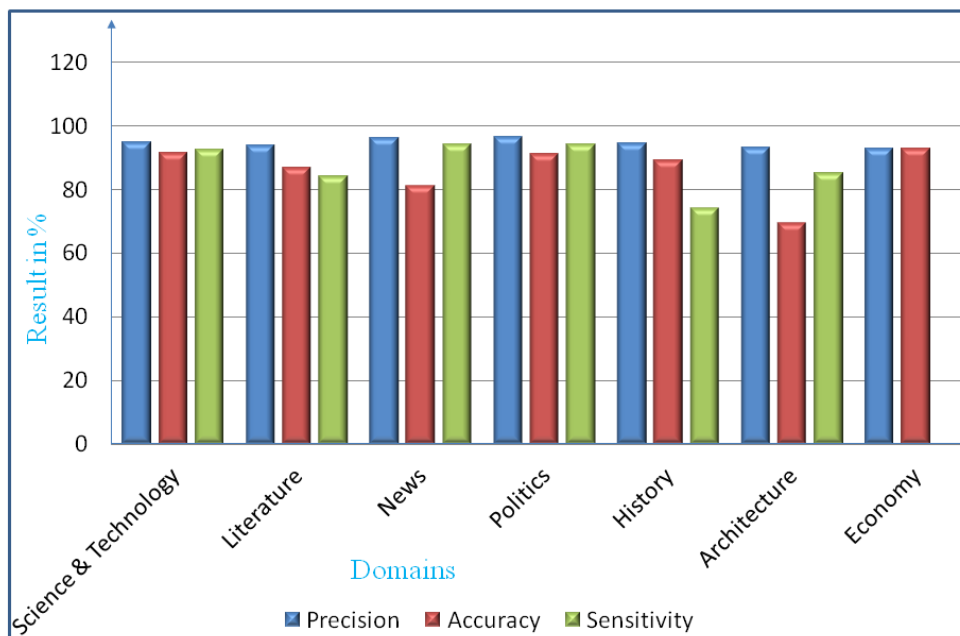


Figure 6.9: Comparative analysis of precision, accuracy and sensitivity for various genre.

(2006)], [Shrivastava and Bhattacharyya (2008)], [Singh et al. (2006)] and [Dalal et al. (2007)] is shown in Table 6.9.

To the best of our knowledge, achieved precision through this system are highest with good precision while keeping data set of 11, 600 words for training and 2, 862 words for testing.

Additionally we have also generated two new special tags for “TIME” and “DATE” compared to the best IIIT Hyderabad tag set [IIITH], which has only 27 tags.

6.5 Summary

In this chapter, we present the automatic POS approach for Devanagari Hindi text. POS tagging method is designed with the help of probabilistic approach and rules based approach.

A manually developed data set of Devanagari Hindi is used to train the system. The method work in two phases, in first phase it tags known words according

Table 6.9: Comparative study of our POS tagger with other Hindi taggers.

Method Used	Model Used	No. of Tags	Training Set (No. of Words)	Testing Set (No. of Words)	Accuracy / Precision
Part-of-Speech Tagging of Hindi Corpus [Modi and Nain (2016)]	Rule-based Model	29	9,000	3,189	91.84 %
Rule-Based Hindi Part of Speech Tagger [Garg et al. (2012)]	Rule-based Model	30	18249	26149	87.55%
Hindi POS Tagging and Chunking [Aniket Dalal and Shelke (2006)]	Maximum Entropy Markov Model	29	35000	8750	88.4%
Hindi POS Tagger Using Naive Stemming [Shrivastava and Bhattacharyya (2008)]	Hidden Markov Model	NA	53400	13500	93.12%
Morphological-Constructing a POS Tagger for Hindi [Singh et al. (2006)]	CN2 Algorithm	27	15562	3890	93.45%
Building Feature Rich POS Tagger [Dalal et al. (2007)]	Maximum Entropy Markov Model	27	15,562	3890	94.38%
Our Approach	Hybrid Model	29	11,600	28,62	95.28%

to trained data set and in second phase it tags unknown words using rule-based approach.

To tag unknown word, various Devanagari Hindi grammar rules, rules based upon prefix, suffix and rules-based on regular grammar are applied. All the rules are derived by calculating probabilities of current word, previous word and next word in the database.

An ambiguity removing module is also provided in the approach. This module is also developed using rule set. This module assigns or reassigns unambiguous tag to the word based upon its contextual information.

Various data sets have been chosen for experiments. These data sets are chosen from seven categories and nineteen sub-categories and also follows Gaussian distribution. The Devanagari Hindi text are tagged by presented approach with average precision of 95.28%, average accuracy of 84.56% and average sensitivity of 88.73%.

Chapter 7

Chunking

Once a corpus is POS tagged, it is possible to bring these morpho-syntactic categories into higher level syntactic relationships with one another, that analyse the sentences in a corpus into their constituents. It is known as partial parsing or chunking. Chunking or shallow parsing is the task of identifying and labeling the simple phrases in a sentence. Chunking is considered a preprocessing step for full parsing. It is also used in limited contexts of identifying noun phrases in named entity recognition [Baskaran (1998)]. In the Chunking process, chunks are identified in given text, in which a chunk can be characterized as a non-recursive, partial structure consisting of inseparable, correlated entities, such that dependencies of the intra-chunk are not distorted [Smriti Singh and Sarma (2012)]. The task of identifying chunk boundaries and chunk labels is modeled in the same way as of identifying POS tags [Dandapat (2009b)]

7.1 Introduction

Chunking is an imperative component for various NLP applications. Corpus is basically a large collection of recorded remarks about text of a particular language which is stored in structured form. Corpus annotation is the procedure of assigning interpretative, linguistic information to the corpus of a particular language. Annotation is generally final result of corpus annotation process; means information is assigned to corpus, which is language dependent. Chunking is one of the various levels of corpus annotation.

A Chunking process involves splitting sentences into non-overlapping segments on

the basis of very superficial analysis. That discovering the main constituents of the sentences and their heads. Which determine syntactical relationships such as subject-verb, verb-object etc. It can be used for information retrieval systems, information extraction, text summarization, bilingual alignment etc. In addition of these, it is also used to solve computational linguistics tasks such as disambiguation problems [Rinku et al. (2014)].

Chunking or shallow parsing or partial parsing is the process of identifying simple phrases (partial structures) in given text. These phrases consist of inseparable, correlated entities and can be of various types like noun phrase, verb phrase, adjective phrase, adverb phrase etc. In the process of chunking generally POS tagging information is considered as contextual information.

Chunking can be formally defined as, given a meaningful sequence $V_1 \cdots V_n$, of the text, the system has to assign chunk sequence $C_1 \cdots C_n$ to the input text, where $V_1 \cdots V_n$ is combination of word sequence $W_1 \cdots W_n$ and respective POS tag sequence $t_1 \cdots t_n$.

Parsing is generally an intermediate step in language processing task. Chunking is completely different from full parsing as it results into simple shallow tree which is less complex from complete parsing tree. It is usually preferred on full parsing or syntactic analysis. Shallow parsing or chunking is preferred as compared to full parsing because it is fast, robust, efficient, easy to implement, less expensive and less complex as compared to full parsing. Shallow parsing can be further used for various NLP tasks like name entity recognition, searching, machine learning, machine translation, information extraction etc. Examples of chunking for POS tagged Devanagari script text are:

Input Part-of-Speech Tagged of Hindi Text:

केंद्र_NNC सरकार_NN ने_PREP 2014_QFNUM में_PREP आयोजित_JVB
 होने_VNN वाली_PREP सिविल_NNC सेवा_NN परीक्षाओं_NN से_PREP सभी_QF
 श्रेणी_NN के_PREP उम्मीदवारों_NN के_PREP लिए_PREP दो_QFNUM
 अतिरिक्त_JJ प्रयासों_NN के_PREP लिए_PREP हाल_NN ही_RP में_PREP
 मंजूरी_NN दी_VFM थी_VAUX ।_PUNC

Chunking of POS tagged Text:

(केंद्र_NNC)_NG (सरकार_NN ने_PREP)_NG (2014_QFNUM में_PREP
)_NG (आयोजित_JVB)_JJG (होने_VNN वाली_PREP)_VG (सिविल_NNC
)_NG (सेवा_NN)_NG (परीक्षाओं_NN से_PREP)_NG (सभी_QF श्रेणी_NN
 के_PREP)_NG (उम्मीदवारों_NN के_PREP लिए_PREP)_NG (दो_QFNUM
 अतिरिक्त_JJ प्रयासों_NN के_PREP लिए_PREP)_NG (हाल_NN ही_RP
 में_PREP)_NG (मंजूरी_NN)_NG (दी_VFM थी_VAUX ।_PUNC)_VG

7.2 Existing Work

In the variety of NLP applications chunker or shallow parser has emerged as an important component. A chunker identifies simple or non-recursive noun phrases, verb groups and simple adjectival and adverbial phrases in running text.

[Akshay and Sangal] presents Hidden Markov Model based chunker for Hindi. They achieved precision of 92.63% for chunk boundary identification task and 91.70% for the composite task of chunk labeling.

[Aniket Dalal and Shelke (2006)] developed a statistical chunker for Hindi language using Maximum Entropy (ME) approach. Chunking is the next step after POS tags are identified. Which involves dividing sentences into nonoverlapping nonrecursive phrases. It serves as first step for full parsing.

In this system, there are six different kinds of chunk labels, like noun phrase (NP), verb phrase (VG), adjective phrase (JJP), Adverb phrase (RBP), conjunct phrase (CP) and others (BLK). The procedure for identifying chunks and their labels is modeled in the same way as that of identifying POS tags. They yield the performance of 86.45% with best accuracies of 87.39%.

A chunking system is developed by [Baskaran (1998)] for Hindi language based

on HMM-based approach in combination with probability models. They achieved 76% precision for chunking.

[Asopa et al. (2016)] developed a rule based chunker for Hindi language. In this rule based approach, the linguistic rules are applied to input text segments. The input is the HMM tagged file to the chunker. For evaluating performance of Chunker, a test corpus of 500 tagged sentences were generated. The system achieved the performance in terms of precision is 79.68%.

According to [Ashish and Sinha] a chunker identifies simple verb-phrases, noun-phrases, adjectival and adverbial phrases in running text. They took advantage of the tool support of the TnT for the chunking of Hindi. By simply feeding (POS-tag, Chunk-tag) to this utility they generate the model for chunking of Hindi. In this approach they yield performance in terms of precision as 89.02%.

A local word-grouping techniques for Hindi language are developed by [Pradipta Ranjan (1999)]. To identifying these word groups correctly and maximally in such languages, reduces the overhead on the core parser for the computational model.

[Avinesh and G] create a chunker using Conditional Random Field (CRF) approach and Transformation based Learning (TBL) for Hindi, Telugu and Bengali. A CRF is a probabilistic model for classification and segmenting data. It define conditional probability distributions $P(Y | X)$ of label sequences given input sequences. The performance they have achieved for Hindi are 80.97%.

7.2.1 Chunking for Indian Languages

Natural Language Understanding (NLU) is valuable field of Artificial Intelligence (AI) and concerned with the speech and language understanding between human and computer.

Phrases identification or chunking is an imperative step in NLU [Bindu.M.S and Idicula (2011)]. Chunking is one of the fundamental processing step for any language processing tool and the most well studied problems in the field of NLP.

A robust chunker is an substantial constituent for various applications requiring NLP. Various techniques have already been tried to automate the task of chunking for English and other languages.

Chunking is essential for many NLP tasks such as identification of structure, information extraction, parsing and phrase based machine translation system. The

Chunker divides a sentence into its major-nonoverlapping phrases and attaches a label to each chunk [Smriti Singh and Sarma (2012)].

Gujarati is one of the primary languages spoken in the western region of India. [Patel and Gali (2008a)] proposed statistical models (Conditional Random Fields) based chunker for Gujarati language to identify the chunks. In which data has been taken from the Central Institute for the Indian Languages (CIIL) corpus, which is freely available for research purpose. The corpus is well distributed among different domains like health, societal, education etc. The IL tagset for chunker consists of 11 different tags and achieved 96% of accuracy.

A text chunker for Indian languages namely Hindi, Bengali and Telugu is developed by [Pattabhi Rao and Sobha (2015)]. Here Transformation Based Learning (TBL) method are used for chunking. The chunker achieved 73.80%, 65.28% and 50.38% accuracy for Hindi, Bengali and Telugu respectively.

A rule based chunker for Bengali is proposed by [Sivaji Bandyopadhyay and Halder (2006)]. The chunker was tested on the unannotated test set after POS tagging and achieved 81.61% accuracy in chunk boundary identification and chunk labeling.

[V. et al. (2009)] developed a chunker for Tamil based on machine learning techniques using MBT, SVM and CRF. They achieved an accuracy of 97.49% for known words and an accuracy of 95.25% for unknown words.

The Manipuri is the Language of north-eastern parts of India, widely spoken in the state Manipur, and a few in the countries of Myanmar and Bangladesh. The Manipuri Language belongs to a high agglutinative class of language. [Kishorjit Nongmeikapam and Bandopadhyay (2014)] developed a chunker for Manipuri language using CRF model with precision of 77.36%.

[Dandapat (2009b)] developed a Indian language (Bengali, Hindi and Telugu) chunker using Maximum Entropy Model with an accuracy of 84.45%, 79.88% and 65.92% for Bengali, Hindi and Telugu respectively.

[Dipanjan Das and Basu (2005)] developed a chunker for Bengali using greedy approach with a precision of around 96%. A Hidden Markov Model (HMM) based chunker for Punjabi is proposed by [Chandan Mittal and Singh (2015)] with an accuracy about 82%.

[Bindu.M.S and Idicula (2011)] developed chunker for Malayalam using AI and rules-based method with 96% precision. A text chunker for Malayalam using

Memory-Based Learning (MBL) approach are proposed by [Rekha and Reghu (drum)] with 97.14% accuracy.

Punjabi is spoken in Punjab region of Pakistan and India. It is also spoken in Himachal Pradesh, Haryana and Delhi and some countries abroad. It is written in two different scripts called Gurmukhi and Shahmukhi. [Jain and Kaur (2015)] proposed chunker for a Indo-Aryan language Punjabi and achieved precision of 93%.

[Ram and Lalitha (2004)] developed a chunker for Agglutinative Language Tamil using FSA (finite state automaton) method. They achieved precision of 94.9%.

[Sankar and Garain (2011)] developed chunker for Bangla language using rules based method and achieved precision of 95.05%.

A Noun Phrase Chunking for Marathi language are proposed by [Sachin Pawar and Bhattacharyya (2015)] using Distant Supervision method. They yield precision of 88.31%. Comparison of chunking for Indian languages are summarized in Table 7.1.

7.2.2 Chunking for Hindi Language

Natural languages are generally categorized into fixed word order languages like English and free word order Indo-Aryan language as Sanskrit, but some Indo-Aryan languages like Hindi have partially lost the free word ordering in the course of evolution. In Hindi word-groups are free-order but the internal structure of word groups are fixed-order [Pradipta Ranjan (1999)].

A HMM based chunker for a relatively free word order language Hindi with simple morphology is present by [Akshay and Sangal]. They yield the precision of 91.70%.

[Aniket Dalal and Shelke (2006)] developed a Maximum Entropy Approach based chunker for Hindi with accuracy of 87.39%. [Baskaran (1998)] developed a Hindi chunker using HMM-based approach in combination with probability models with 76% precision.

A rule-base chunker for Hindi language are developed by [Asopa et al. (2016)]. They yield the precision of 79.68%. [Ashish and Sinha] proposed a generalized, approach for robust Hindi chunker based on HMM with 89.02% precision.

Table 7.1: Comparative study of chunking for Indian languages.

Method Used	Model Used	Accuracy/Precision
Chunker for Gujarati [Patel and Gali (2008a)]	Statistical models(CRF)	96%
Indian language Chunker Pattabhi Rao and Sobha (2015)]	TBL	73.80%.
Bengali chunker [Sivaji Bandyopadhyay and Halder (2006)]	Rule-based	81.61%
Tamil Chunker [V. et al. (2009)]	MBT, SVM, and CRF	95.25%
Manipuri Chunker [Kishorjit Nongmeikapam and Bandopadhyay (2014)]	CRF	77.36%
Indian language Chunker [Dandapat (2009b)]	Maximum Entropy	84.45%
Bengali Chunker [Dipanjan Das and Basu (2005)]	Greedy Approach	96%
Chunker for Punjabi [Chandan Mittal and Singh (2015)]	HMM	82%
Malayalam Chunker [Bindu.M.S and Idicula (2011)]	AI and Rule- based	96%
Chunker for Tamil [Ram and Lalitha (2004)]	FSA	94.9%
Chunker for Malayalam [Rekha and Reghu (drum)]	MBL	97.14%
Punjabi Chunker [Jain and Kaur (2015)]	Greedy based algorithm	93%
Bangla Chunker [Sankar and Garain (2011)]	Rulse based	95.05%
Marathi Chunker [Sachin Pawar and Bhattacharyya (2015)]	Distant Supervision	88.31%.

Table 7.2: Comparative study of chunking for Hindi languages.

Method Used	Model Used	Accuracy / Precision
Hindi Chunker [Akshay and Sangal]	HMM	91.70%
Chunker for Hindi [Aniket Dalal and Shelke (2006)]	Maximum Entropy	87.39%
Hindi Chunker [Baskaran (1998)]	HMM	76%
Hindi Chunker [Asopa et al. (2016)]	Rulse based	79.68%
Hindi Chunker [Ashish and Sinha]	HMM	89.02%
Chunker for Hindi [Avinesh and G]	CRF and TBL	80.97%
Noun and Verb Group Identification for Hindi [Smriti Singh and Sarma (2012)]	CRF and Morphotactical Information	95.26%

[Avinesh and G] proposed chunking method using CRFs and Transformation Based Learning (TBL) for Telugu, Hindi and Bengali. They achieved an accuracy of 79.15%, 80.97% and 82.74% for Telugu, Hindi and Bengali respectively.

[Smriti Singh and Sarma (2012)] proposed noun and verb group method using morphological information and CRF and yield accuracy of 95.26%. Comparison of chunking for Hindi language are shown in Table 7.2 .

7.2.3 Chunker for Other Languages

[Ali and Hussain (2010)] developed a hybrid verb phrase chunking approach for Urdu using HMM based statistical and correction rules. They achieved an accuracy of 95.14%. [Gao et al. (2006)] proposed a chunking method for Chinese using editing support vector machine (ESVM) and K nearest neighbors (KNN) with precision of 84.11%.

A lexicalized HMM-based approach to Chinese text chunking are proposed by [Fu et al. (2005)] and improve the performance of a HMM-based chunking system. [Siddiq et al. (2010)] developed a Urdu Noun-Phrase Chunking method with hybrid approach. In this statistical method (HMM) and hand crafted rules are used

Table 7.3: Comparative study of chunking for other languages.

Method Used	Model Used	Accuracy/Precision
Urdu verb Phrase Chunking [Ali and Hussain (2010)]	Hybrid(HMM and Rulse)	95.14%
Chinese Chunking [Gao et al. (2006)]	ESVM-KNN	84.11%
Chinese text chunking [Fu et al. (2005)]	Lexicalized HMM	90.82%
Urdu Noun Phrase Chunking [Siddiq et al. (2010)]	Hybrid	93.87%
Persian Chunker [Kiani et al. (2009)]	Hybrid	85.7%

and achieved an accuracy of 93.87%. [Kiani et al. (2009)] developed a chunker for Persian language with hybrid method and achieved precision of 85.7%. Comparison of chunking for others languages are shown in Table 7.3.

7.3 Techniques for Chunking

There are various types of techniques for chunking, all these techniques can be divided in two broad categories, like supervised learning and unsupervised learning. Both learning techniques are further divided in three major types of techniques, i.e., stochastic based approach, rule-based approach and hybrid approach. Supervised stochastic approaches cover various models like Hidden Markov Model, memory based model, support vector machine, neural network based model etc [Antony (2011b)].

Where as for unsupervised stochastic approach very few work has been done, because this type of technique only classify the input text but does not provide tags to input text. To assign tags some labeled text must be used.

Supervised POS tagging and Chunking is a machine learning methodology, which uses a pre-tagged, human made corpus as training data. This pre-tagged corpus is required to learn information about probabilities, a word's probable tags, word-tag frequencies, tagset for annotation and rule sets etc. Accuracy of this type of learning increases as the size of training data set increases [Hasan (2006)].

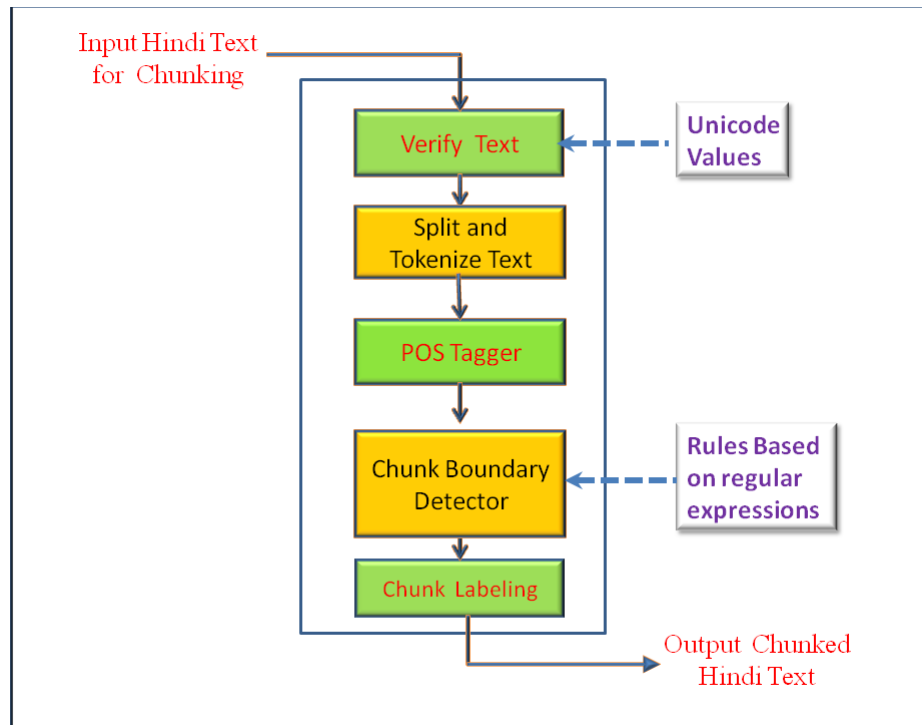


Figure 7.1: Experimental framework of our chunking approach.

Unsupervised POS tagging or Chunking is opposite to supervised methods of POS tagging or Chunking, as unsupervised methods do not use any pre-tagged, human made corpus. This kind of taggers use some advanced computational methods to automatically generate transformation rules and different classes. On the basis of available information and data these type of methods find language dependent rules to calculate probabilities [Hasan (2006)]. These techniques are detailed in previous chapter.

7.4 Our Method for Chunking

In our approach for chunking, a rule-based method is used. The rules are generated by performing statistical analysis over 11,600 words with manually POS tagged corpus of Hindi language. These rules are exemplified by using regular expressions and implemented using various FSM (Finite State Machine) automatas.

The description of chunking categories and rules used in our method are describes in this section.

7.4.1 Categories of Chunking

In our method of chunking, eight categories are used for chunk labeling. In Hindi language, two major types of chunks exist as noun chunk and verb chunk. In addition to these chunks, five other chunk categories are pronoun, adjective, adverb, negative and conjunction.

If the input text are not chunked by our method in any categories as mentioned above, then “MISC” tag is assigned to it. A head word and functional words are contained by all these types of word groups. The functional words are dependent on head words.

1. **Chunks of Noun:** Noun chunks finding is a sub-task of text chunking. It finds the chunks which contains noun phrases. In essence all noun chunks consists of a noun as a head word which is an essential part of a noun chunk. With head word, some modifiers may also be as a optional part of noun chunk. For example:
 - किला (Noun)
 - पुराना किला (Adjective + Noun)
 - वो पुराना किला (Pronoun + Adjective + Noun)
2. **Chunks of Verb:** Verb chunk finding is also a sub task of text chunking. The verb chunks also contain a head word which always is a verb and some other functional words which can be auxiliary verbs or postpositions. For example:
 - अधि (Verb main + Postposition)
 - उन (Verb main + Verb auxiliary)
3. **Other Chunks:** The chunker consists of five more chunking categories other than noun and verb chunks, namely pronoun, adjective, adverb, negative and conjunction. Adverb, adjective and pronoun will form respective chunks only, if they appear independent from any noun text. The negatives and conjunctions form their respective chunks. All the chunk categories are illustrated in Table 7.4.

Table 7.4: Chunk tag set used in the system.

S. No.	Chunk Type	Annotation Convention
1.	Noun Type	NG
2.	Pronoun Chunk	PRG
3.	Verb Chunk	VG
4.	Adjective Chunk	JJG
5.	Adverb Chunk	RBG
6.	Conjunction Chunk	CCG
7.	Negative Chunk	NEGG
8.	Miscellaneous	MISC

7.4.2 Generation of Rules

In rules generation, a manually developed corpus of Devanagari Hindi is used. The corpus consists of 11,600 Hindi words with their respective POS from our corpus tags.

Detailed analysis of corpus has been performed and co-occurrence frequency, entropy and information gain values of POS tags had been generated.

The formula used for entropy and information gain are as follows:

$$Entropy(C) = \sum -p_i \log_2 p_i \quad (7.1)$$

Here C is the corpus, p_i is the part of corpus belonging to class i and total number of classes are equal to total number of POS tags in the tagger.

$$Gain(C, A) = Entropy(C) \sum \left(\frac{|C_v|}{|C|} \times Entropy(C_v) \right) \quad (7.2)$$

Where A = Attributes.

\sum is sum of every value v of all possible values of attributes A.

$C_v \subset C$ for which attribute A has value v.

$|C_v|$ = number of elements in C_v

$|C|$ = number of elements in C.

After that these values have been analyzed to develop the rules.

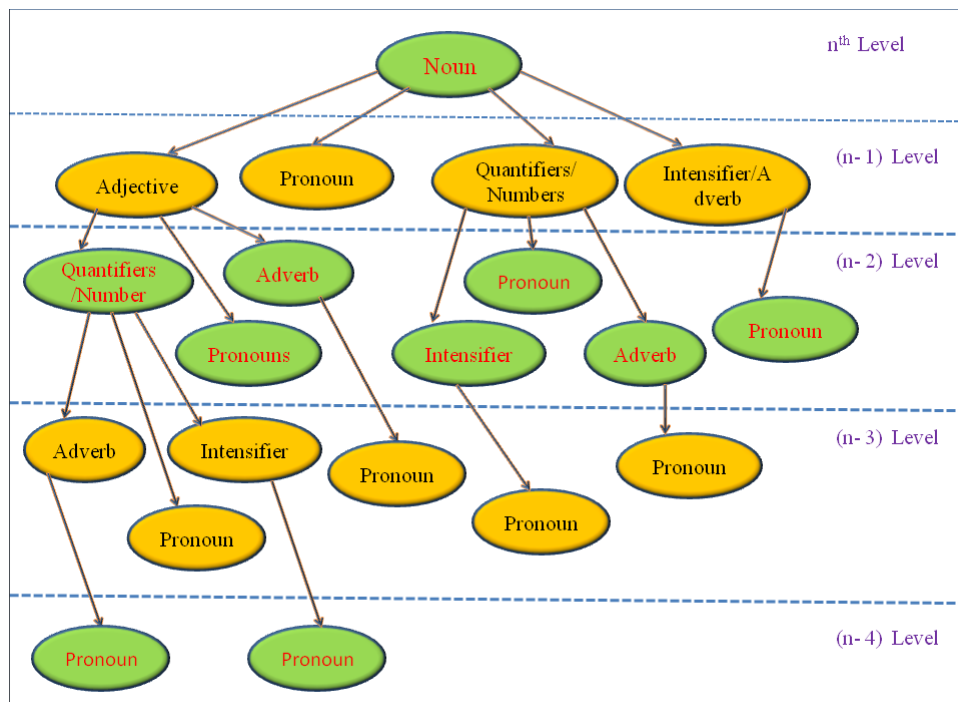


Figure 7.2: Decision tree architecture for noun chunk identification.

7.4.2.1 Rules Generation for Noun Groups

In support of noun chunks identification we generate a sequence of tags which covers noun phrases. The sequence is generated by calculating information gain over POS tags of current words and previous words using ID3 algorithm [Rizvi (2010)]. This algorithm maximizes the information gain. At first information gain is calculated for n^{th} level tags and (n-1) level tags. Then the process is repeated with (n-1) level tags, (n-2) level tags and so on, to generate a complete decision tree for noun chunk identification.

In the process of decision tree generation, at every level one or more than one node are considered which gave considerably high information gain. A complete decision tree of POS tags for noun chunks identification is shown in Figure 7.2.

A noun chunk can be finished at any level of the decision tree. That can be expressed mathematically by using regular expression as shown in Equation 7.3. In equation, optionality is shown by (), and repetition is shown by *.

$$NG = (A1)(A2)(A3) * A4(A5)(A6) \quad (7.3)$$

In the above Equation A1 includes pronoun, A2 includes numerals or intensifier,

A3 include adjectives and all three elements are optional. Here A4 includes noun and called head word. This is the compulsory element of a noun chunk, A5 includes postpositions and A6 includes particles and both are optional in case of a noun chunk.

7.4.2.2 Rules Generation for Verbs Groups

In identification of verb chunk, same procedure is followed as above. A verb chunk start with a verb which is a head word for verb chunk. This verb is followed by auxiliary verbs or postpositions or particles or all to complete verb chunk. The complete regular expression for verb chunk identification can be expressed mathematically by using regular expression as shown in Equation 7.4.

$$VG = A1(A2) * (A3)(A4) \quad (7.4)$$

In the above Equation A1 includes verb and is called a head word. This is the compulsory element of a verb chunk. Here A2 include auxiliary verbs, A3 include postpositions, A4 include particles and all are optional in case of a verb chunk.

7.4.2.3 Rules Generation for Other Groups

The same procedure is followed for generation of a regular expression for other chunks also. If an adjective, pronoun or adverb comes within the regular expression as shown by Equation 7.5. Then they will be grouped within noun group, otherwise they will be grouped as pronoun chunk, adjective chunk or adverb chunk respectively. The method follows separate regular expression for each of these types of chunks.

These regular expressions are as follows:

$$JJG/RGB = (A1)(A2)A3(A4)(A5) \quad (7.5)$$

In the above Equation A1 includes pronoun, A2 includes numeral or intensifier, A3 includes adjective or adverb and called head word. This is compulsory element of adjective or adverb chunk. The A4 includes postpositions and A5 includes particle, and both are optional too in case of an adjective or adverb chunk.

$$PRG = A1(A2)(A3) \quad (7.6)$$

In Equation 7.6, A1 includes pronoun and is called a head word. It is compulsory element of a pronoun chunk. The A2 includes postposition and A3 includes particles, and both are optional element in case of a pronoun chunk. A conjunction and negative chunk finds conjunction or negative words in input text and assign conjunction or negative chunk respectively.

7.4.2.4 Some Special Cases

In our approach for chunking by rule-based techniques, following special cases are also considered:

1. All types of punctuation marks are included in its left chunk by default.
2. All quantifiers are chunked as noun chunks.
3. In case of nouns, verbs or adjective which occur in kriyamulla. These are not taken separately and will be considered as noun chunk, verb chunk or adjective chunk respectively.

7.4.3 Experimental Setup of The Chunker

In our work of Hindi language chunker designing, a rule-based approach is used. The chunking algorithm is divided into two phases, namely identification of chunk boundary and chunk labeling.

7.4.3.1 Chunk Boundary Detector

The chunk boundary detector marks boundaries of various chunk in the input text. The boundary detector works on regular expression basis, as stated in previous section. In this all the regular expression are implemented through different finite state machine (FSM) automaton.

An example of FSM for noun chunks identification is shown in Figure 7.3. Likewise, one FSM automaton is built for each type of chunk. The boundary detector reads the POS tagged data and marks all the candidates in text. Where a candidate will be a word with noun, verb main, adjective, pronoun, adverb, conjunction or negative tag. On the basis of a candidate, its respective FSM automaton is chosen to run.

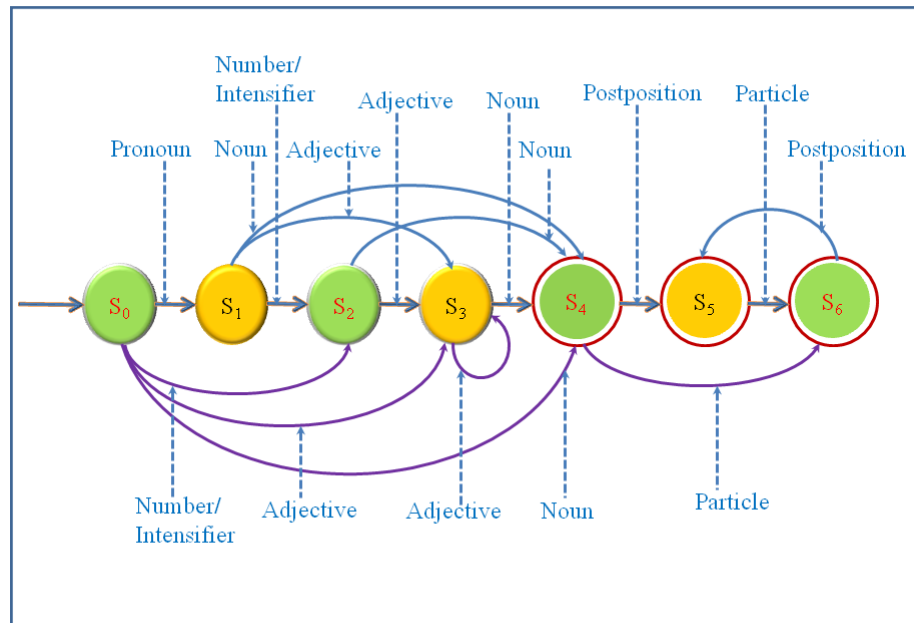


Figure 7.3: Finite machine automaton for noun chunk identification.

On foremost all noun candidate are processed to find noun chunks boundaries because noun chunks can contain adjective, pronoun etc in it. Subsequent to that all other candidates are processed one by one with respective FSM. The FSM automaton will run until it gets to the end of the sentence or some invalid transition. On the basis of FSM automaton, both boundaries of chunks are marked by boundary detector. Finally the chunk boundary detector's output goes for chunk labeling.

7.4.3.2 Labeling of Chunk

The chunk labeling is done after identification of chunk boundaries. In chunk labeling, POS tags are verified within a group boundary. If there exists noun, finite verb, conjunction or negative, then respective chunk tags are assigned to them. In other respects pronoun, adjective and adverb tags are checked and according to the present head word, a particular chunk label is assigned to the chunk groups that are identified in previous section.

7.4.4 Results and Performance Analysis

There are various types of evaluations measures to arbitrate the consequence and quality of the approach, like precision or positive predictive value, sensitivity or recall, true positive rate, false positive rate etc.

In our work three types of evaluation measures are used to verify the approach for chunking. These measure are precision, recall and accuracy as affirmed in Equation 7.7, 7.8 and 7.9 respectively.

$$Precision = \frac{TP}{TP + FP} \quad (7.7)$$

$$Recall = \frac{TP}{TP + FN} \quad (7.8)$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (7.9)$$

Where, TP = True positive,

TN = True negative,

FP = False positive and

FN = False negative.

The system has been validated on various data sets of different domain and size as specified in Figure 7.5. The validation is performed through holdout method of cross validation as discussed in previous chapter. In which complete data set is divided in two different sets, 75% is training set and 25% is the testing set.

The rules are derived from training set and upcoming data are according to these rules. The chunking system is tested on two types of data, as manually POS tagged data and untagged data, which are tagged by our approach. The evaluation result on manually POS tagged data are as follows:

- Total number of chunks in input text = 1539.
- Number of chunks identified by the method = 1539.
- Number of chunks correctly identified by method = 1521.
- Precision of the system = 98.83%.
- Accuracy of the system = 98.83%.
- Sensitivity of the system = 100%.

Table 7.5: Performance result of chunking approach in terms of P-Precision, A-Accuracy and S-Sensitivity.

Domains	No. of Words	No. of Chunks	No. of Chunks Identified	Correctly Identified Chunks	P	A	S
Science & Technology	340	160	141	136	96.45	85.00	88.12
Literature	500	243	205	190	92.68	78.18	83.67
News	662	310	280	270	96.42	87.09	90.32
Politics	516	240	220	216	98.18	90.00	91.66
History	382	174	148	139	93.91	79.88	85.05
Architecture	226	115	105	100	95.23	86.95	91.30
Economy	236	125	114	109	95.61	87.34	91.20

On the untagged data (POS tags are assigned by our approach) various genres are considered as Science & Technology, Literature, News, Politics, History, Architecture and Economics. The holdout method are also used here for cross validation. Chunking results are summarizes in Table 7.5. Results achieved by the method in graphical form for different domains shown in Figure 7.6.

The system achieved 95.49%, 84.92% and 88.76% precision, accuracy and sensitivity respectively in average. The best precision, accuracy and sensitivity are 98.18%, 90.00% and 91.66% respectively for politics domain. The system shows 92.68%, 78.18% and 83.67% lowest results for precision, accuracy and sensitivity respectively in Literature domain.

Error analysis of each chunk type are shown in Figure 7.7. We can see that error for noun chunks identification is 8.9%, which is highest, as noun is the most frequent tag in the system. Error for conjunction and negative chunk is lowest which is zero, as both chunks are identified with 100% precision and accuracy. The error for other chunks is also proportional to their respective occurrences in the corpus.

The comparative analysis of our chunker with state-of-art Hindi chunkers such as [Akshay and Sangal], [Aniket Dalal and Shelke (2006)], [Baskaran (1998)], [Asopa et al. (2016)], [Ashish and Sinha], [Avinesh and G] and [Smriti Singh and Sarma (2012)] is shown in Table 7.6.



Figure 7.4: Sample result of chunking.

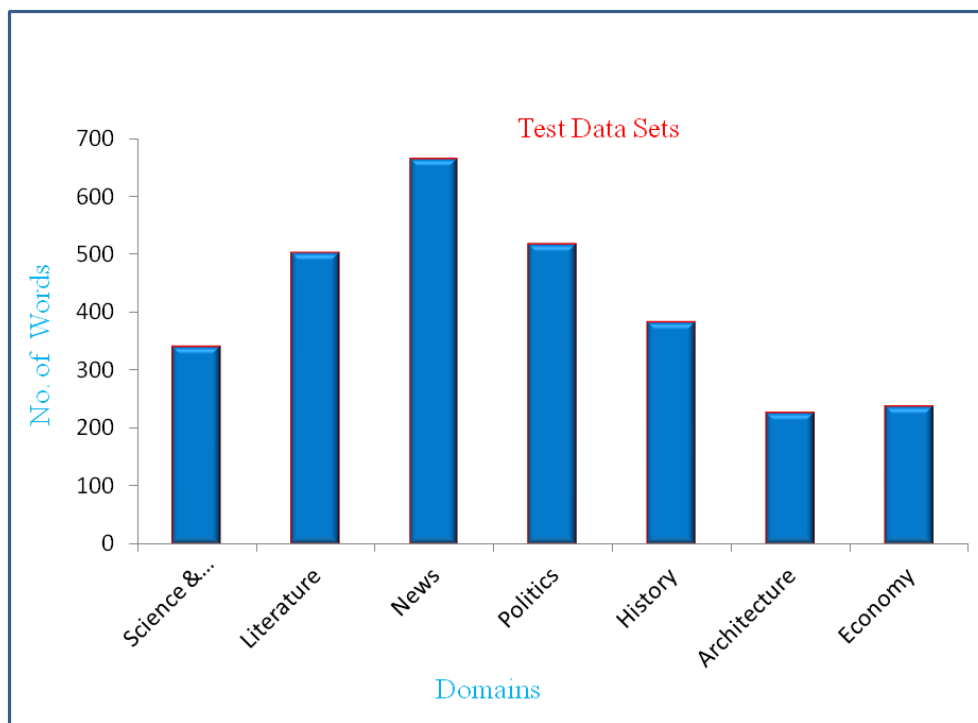


Figure 7.5: Data set of different domains for testing.

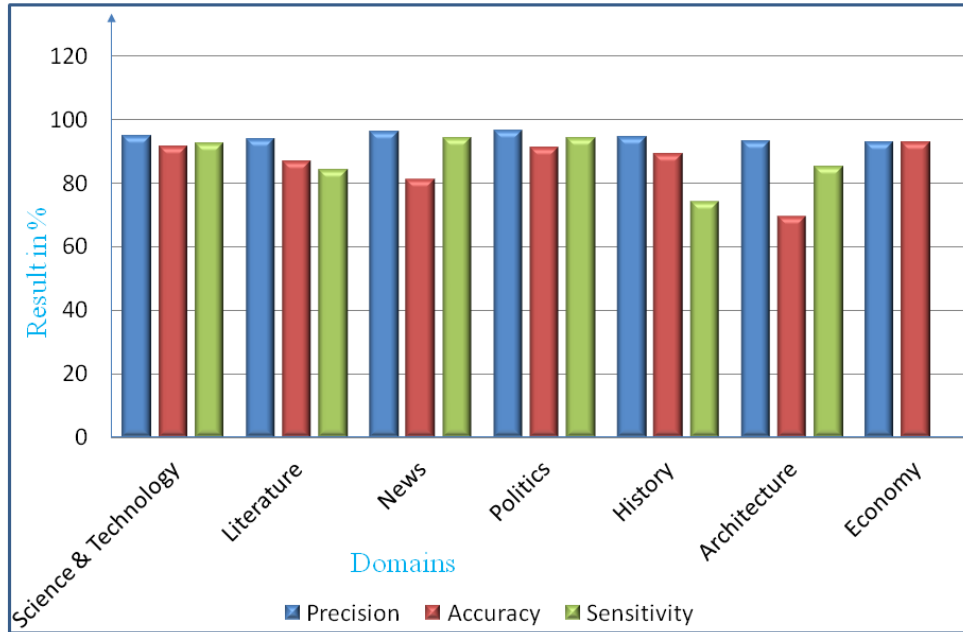


Figure 7.6: Comparative analysis of precision, accuracy and sensitivity for various genre in chunking.

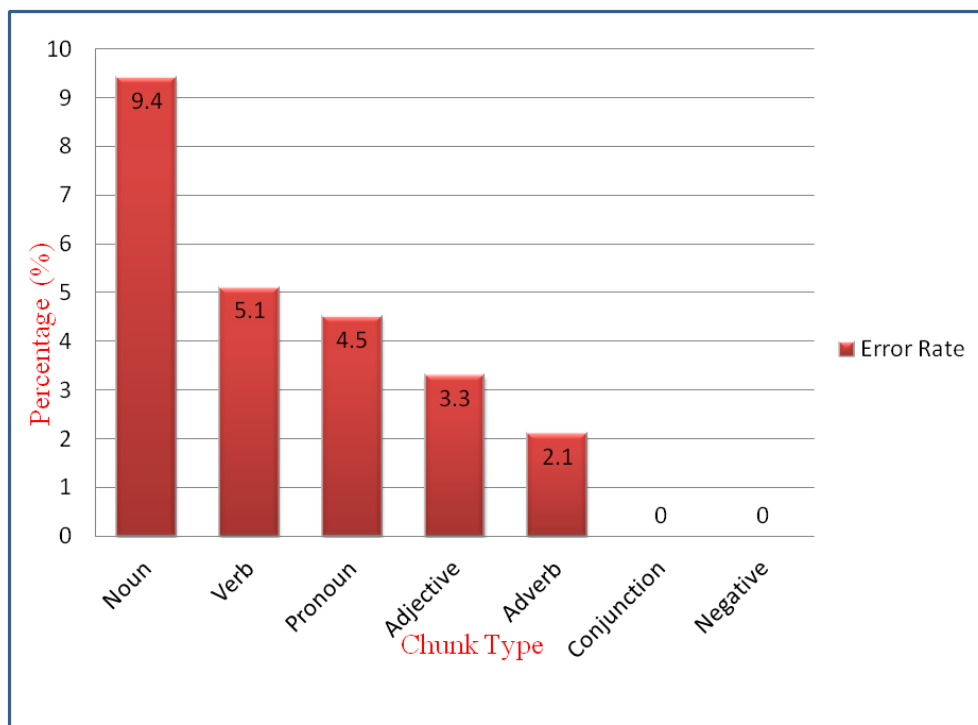


Figure 7.7: Per chunk tag error analysis.

Table 7.6: Comparative study of chunking models with our chunker.

Method Used	Model Used	Accuracy / Precision
Hindi Chunker [Akshay and Sangal]	HMM	91.70%
Chunker for Hindi [Aniket Dalal and Shelke (2006)]	Maximum Entropy	87.39%
Hindi Chunker [Baskaran (1998)]	HMM	76%
Hindi Chunker [Asopa et al. (2016)]	Rulse based	79.68%
Hindi Chunker [Ashish and Sinha]	HMM	89.02%
Chunker for Hindi [Avinesh and G]	CRF and TBL	80.97%
Noun and Verb Group Identification for Hindi [Smriti Singh and Sarma (2012)]	CRF and Morphotactical Information	95.26%
Our Approach	Rule-based Model	95.49% (Precision)

7.5 Summary

In this chapter, we present an automatic chunking approach for Devanagari Hindi text. The chunking method does not use any manual chunked corpus of Hindi language. Approach uses only the POS tag information of the text as contextual information. The approach is based on rule-based model. These rules are generated by detailed analysis of POS tagged corpus of Devanagari Hindi language (our) and represented through regular expressions. Various FSM automaton are used to implement these rules.

Various data sets have been chosen for experiments. These data sets are chosen from seven categories and nineteen sub-categories. The system achieved 95.49%, 84.92% and 88.76% precision, accuracy and sensitivity respectively an average.

The best precision, accuracy and sensitivity are 98.18%, 90.00% and 91.66% respectively for politics domain. The system shows 92.68%, 78.18% and 83.67% lowest results for precision, accuracy and sensitivity respectively in literature domain.

Chapter 8

Statistical Analysis of Corpus

The corpus authenticity and applicability count on the feature of the texts and optimality of texts in the corpus. In the recent era of digitization conjecture compilation of corpora made it even more apparent and revealed many fundamental research areas in the language processing domain.

In this chapter, we have detailed the statistical techniques used for analysis of the data of our corpus. The assessment of the data has been carried out against the outcomes achieved by the DSHTC (Devanagari Script Handwritten Text Corpus) framework.

8.1 Introduction

We explain the statistical analysis of DSHTC corpus implying on corpus texts and frequency of occurrences. The perception of statistical analysis is conducted utilizing integral statistics such as rank of words, frequency of words, entropy and perplexity of the corpus. In addition to these conventional corpus statistics, some of the robust features based statics such as utilizing Zipfs linguistic law are also tested. Second the measurement of dispersion in corpus information using pure frequency profiling, standard deviation and coefficient of variation. The experimental results obtained from statistical analysis observation are to vindicate viability and usability of corpus data.

8.2 Study of Corpus Statistics

The statistical analysis of many existing corpora for developing standard language models and the applicability of the corpus in research eminence are embarked on by researchers working in corpora domain. In statistical analysis of language corpora various unique terms are utilized. These standard terms help in building a canonical mathematical model for languages.

[Goyal and Lalit (2011)] performed statistical analysis of transcript Hindi text and compared with Punjabi text. [Bharati Akshar and S.M. (2002)] presented fundamental statistical analysis for ten machine-readable Indian languages corpora and discussed the comparative study results. [Agrawal SS. and Minakshi (2014)] presented statistical analysis of Punjabi, Hindi and Nepali to calculate the information about entropy, perplexity etc.

[Majumder and Khan (2006)] describes the aggregating approach of Bangla News corpus namely “Prothom-Alo” along with the statistical perception analysis of texts.

[Nemeth and Zainko (2002)] describes a multilingual statistical analysis of texts using through parallel study of three different languages Hungarian, German and English. Besides the statistical observation, also provide some comparative results signify this study.

[Gries (2008)] presents the corpus linguistic analysis predicated on frequency occurrence and co-occurrence of frequency. Also discussed various measure of dispersion for deviation of proportions, with conceptually simpler and more versatile as compared to existing ones.

8.3 Statistical Measurement of Corpus

The least complex and most persistent statics as a part of the corpus for authentic analysis is the linguistic variable that may be word token, tag of word etc. The apparent way of computation of corpus linguistic is analyzing the statistics, that characterize the data. In our corpus for DSHTC calculates following statistics:

1. The frequencies of occurrence of a linguistic variable known as the observed frequencies, that might be normalized frequencies in percent, per thousand words, per million words.

2. The ranks of the linguistic variable are computed from frequencies as Type /Token Ratio (TTR), vocabulary, and grammar information in given language.
3. The frequencies of affiliation measures that do not include actual enormousness testing like mutual information; that evaluate the relationship of one linguistic variable to another linguistic variable which might be a word or a syntactic pattern.
4. Measures of central tendencies, for example, means or medians.
5. Dispersion measures, in statistics measure how spread out an arrangement of information from a center point is. A point when an information set has an exhaustive scattering, when scattering is little the things in the set are immovably grouped. Dispersion measures accompany averages, such as standard deviations, standard errors etc.

8.3.1 Frequency Profiling

Our corpus framework DSHTC also perform the frequency profiling of the words. Hence, in this section we have focused on statistical analysis based on frequency profiling. Information about the words frequency distribution is substantial compared to selection of texts and vocabulary rank.

The frequency statistics is also valuable in the field of language analysis, language teaching and enlargement in language vocabulary. Besides these domains, it is also required in frequency based word list in NLP.

The DSHTC generates a word frequency profile from an annotated text corpus using the tokenization of the word, using the interface, which is supporting inbuilt query structure. That generates the list of words and their frequency of occurrence.

The complete process of generating a frequency list from the data set are shown in Figure 8.1. In respective to word in the data set, it verifies whether the word exists in our database so far; if it is there frequency of that respective word is increased by one, and if the word is not there, add it to the list and assign it a frequency of one.

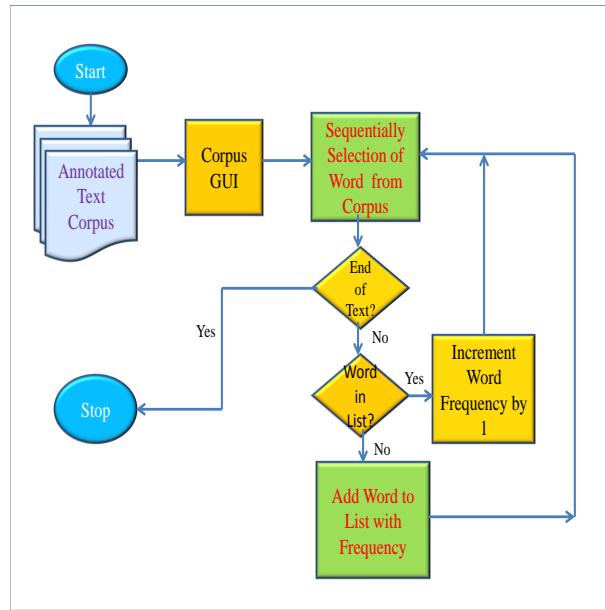


Figure 8.1: A complete flow process of frequency generation of words in our corpus.

8.3.2 Entropy and Perplexity Measurement

Entropy is used to quantify uncertainty associated with a random variable. In NLP and Computational Linguistics, entropy is used as the quantification of information. It finds application in various fields, like how well descriptive linguistic matches a given language, how much information is there in a specific descriptive linguistic data etc. The information is more prognostic-able for the low entropy than the higher entropy. Let the probability of occurrence of a word in corpus is $p(x)$, and is defined as:

$$p(x) = \frac{\text{frequency of word}}{\text{Total word Tokens}} \quad (8.1)$$

The entropy on the basis of probability of occurrence as shown in Equation 8.1, is defined as:

$$H(X) = \sum_{x=1}^X -p(x) \log_2 p(x) \quad (8.2)$$

Here X is the total number of words in the corpus, and x is random words from one to the range of word types within the corpus of data collection. The most intuitive approach to characterize entropy is to discover the number of bits it would require in encoding a certain piece of information.

Likewise, relative entropy, maximum entropy, redundancy and perplexity have been also calculated for DSHTC.

H-Maximum or maximum entropy obtained when the probabilities of all words in the corpus are same. Where H-actual is entropy obtained from the formula described above. H-maximum is acquired when the probabilities of the considerable number of words in the corpus are same. Then H-maximum would be as follows:

$$H(\text{maximum}) = \log_2 N \text{ (Where } N \text{ is the word types)} \quad (8.3)$$

Similarly, relative entropy or relative uncertainty can be calculated using the following formula

$$H(\text{relative}) = \frac{H(\text{actual})}{H(\text{maximum})} \quad (8.4)$$

$$\text{Redundancy} = \frac{H(\text{maximum}) - H(\text{actual})}{H(\text{maximum})} = 1 - H(\text{relative}) \quad (8.5)$$

The proportion of redundancy may be varied from acclimated to investigate depending on whether or not the data may be compressed and kept in less number of bits or not respectively.

Perplexity is useful for evaluating language models. Perplexity for the given Corpus data can be evaluated by:

$$\text{Perplexity } P(X) = 2^{H(X)} \quad (8.6)$$

Where the perplexity is the measurement in information theory, in NLP, perplexity is a prevalent way of evaluating language models. It measures the goodness of the model. Lower perplexity denotes the corpora is more prognosticable than the higher perplexity

Comparative analysis of Entropy, H-maximum, Redundancy and Perplexity of Indic scripts corpus with our corpus are shown in Table 8.1.

Table 8.1 illustrates the comparative values obtained for the defined parameters for the our Devanagari script Hindi corpus and other Indic corpora. From the Table 8.1, can be observed that the Devanagari Hindi language have high redundancy values in comparison to size of other Indic corpus. The value of Entropy for the DSHTC model for Devanagari Hindi corpus indicates that, each word requires

Table 8.1: Comparative analysis of Entropy, H-maximum and Redundancy of Indic scripts corpus and our corpus.

Scripts	Words	Entropy	H-max Entropy	Relative Entropy	Redundancy
Assamese	189,860	13.11	17.53	.747	.253
Bengali	190,841	13.02	17.54	.742	.258
Hindi	127,241	11.088	16.95	.654	.346
Kannada	346,850	14.88	18.40	0.808	.192
Marathi	210,578	13.56	17.68	0.766	.234
Oriya	158,903	12.99	17.277	.75	.25
Punjabi	102,255	11.38	16.64	.683	.317
Telugu	633,169	15.625	19.27	0.81	.19
Tamil	452,881	15.21	18.78	0.809	.191
URDU	46,664	10.689	15.51000	.6885	.3114
Our corpus	123,750	10.8962	15.6283	.7136	.2863

an average 10.8962 bits for encoding.

Applying the above formulas the most extreme entropy and redundancies are figured for the corpus. The statistically measurable information serves to locate the number of bits needed to encode a word in a given language and redundancies rate for every language. The redundancies rates can be used to break down whether the information can be compacted and put away in less number of bits or not nearly. From Table 8.1, it can be observed that the Hindi language have high redundancy values. Thus, data compression is more expensive for Hindi languages than other Indian languages.

It can also be observed from Table 8.1, that if we compare the values of both words and perplexity then, lower perplexity values of our Devanagari script Hindi corpus indicates the goodness of model in terms of quality of data collection.

8.3.3 Degree of Dispersion Measurement

Dispersion has a specialized importance in statistics. The dispersion measures the focal point of the information and also a perspective perception. Another element of the perception is how the perception spreads about the middle. The perception may be near the center, or may be spread far from the middle. In the event that the perception is near the center, we say that dispersion, scatter or variation is

small. On the off chance that the perception is spread far from the middle, we say dispersion is enormous.

In measuring dispersion, it is essential to know the measure of variation and the level of variation. There exists a substantial number of dispersion measures strategies; here we described a percentage of the broadly known techniques on our database to figure out the scramble information about the corpus data.

8.3.3.1 Dispersion Measure using the Frequency Profiling

Frequency profiling depicts a procedure to figure out the degree of dispersion in the linguistic data set with the use of frequencies of occurrence of linguistic variables. The most fundamental corpus-based statistic is the observed frequency of some phenomenon. Word frequency distributions have been concentrated intensively from both abstract and linguistic viewpoints.

The measurement of the degree of dispersion can quantify the counted linguistic expression to specify that how well the corpus frequency reflects the expressions of the overall distribution of data in the corpus.

We have normalized the scattering, this normalized dispersion (DPnorm) measure ranges from 0 to 1, where values near zero imply that the relative frequencies of occurrence of the linguistic variable expression in the corpus are straightforwardly corresponding to the sizes of the corpus parts while values close to 1 means that the variable linguistic expression is distributed unevenly among the corpus parts.

The criteria we have applied for selection words, which can represent the nature of the whole corpus to figure out the scramble information about the corpus data is described in Algorithm 1. The algorithm first ensure that, frequency of each input word should be greater then or equal to 10. Later, for selection of words in a frequent distance to make representative of the whole corpus. We applied a nested for loop, where each time step 3 gives a frequency and nested loop at step 4 retrieves 15 words with similar frequency from the corpus. In addition to outcomes obtained from the H, we have also selected the five most highest and lowest frequency words having a frequency greater than ten.

The results obtained in the above steps is the collection of hundred selected words. We applied the below mentioned frequency profiling techniques to find out the

Algorithm 1 Selection of 100 random words.

Ensure: $f_w \geq 10$ {where f_w is frequency of words}

Step:1 $T = \log_{10} \frac{HF}{n}$ {where HF is highest frequency}

Step:2 for $i \in \{1, \dots, n\}$ {where n is part of corpus}

Step:3 $S_i = (5)^iT$

Step:4 for $j \in \{1, \dots, 15\}$

Step:5 $A = \{w_j \mid (w_j \in C) \cap (f(w_j) \approx S_i)\}$ { where w_j is a word and C is the Corpus}

degree of dispersion in corpus information for these selected words.

A measure of dispersion for a selected word in our corpus (DP / DP-norm) is computed using frequency profiling is defined in Algorithm 2.

Algorithm 2 Calculation of dispersion and normalize-dispersion .

Ensure: Selected words from corpus.

Step:1 Calculate the Relative or Observed Frequency.

$RF = \frac{f(w_p)}{f(w_c)}$ {here, $f(w_p)$ is frequency of word in each part and $f(w_c)$ is frequency of word in total corpus}

Step:2 Calculate Expected Frequency (EF).

$EF = \frac{T(w_p)}{T(w_c)}$ { here, $T(w_p)$ is the total number of word in each part and $T(w_c)$ is total number of word in corpus}

Step:3 Absolute difference for each part.

Step:4 for $i \in \{1, \dots, n\}$

$Abs(n_i) = |(R.F.) - (E.F.)|$

Step:5 Summations of Absolute difference.

$Sum = \sum_{i=1}^n Abs(n_i)$

Step:6 DP value:

$DP = \frac{Sum}{2}$

Step:7 Normalized DP value is as:

$DP_{norm} = \frac{DP}{1 - \min(s_i)}$

where s_i is the Expected Frequency (EF) of smallest corpus part. The reason for selection of s_i for normalization is to resolve the issue of unexpected value of DP, when corpus is not divided in equal part.

Value of the normalized Dispersion (DPnorm) is close to 0 when a word is distributed evenly, and close to 1 when a word is distributed unevenly.

Figure 8.2 demonstrates the relation between frequency and Degree of Dispersion on the premise of the chosen hundred words from the diverse frequency of the corpus data. The resulted DP values varies between 0.04 to 0.69. From one viewpoint, there is a probabilistic connection between the frequencies of components

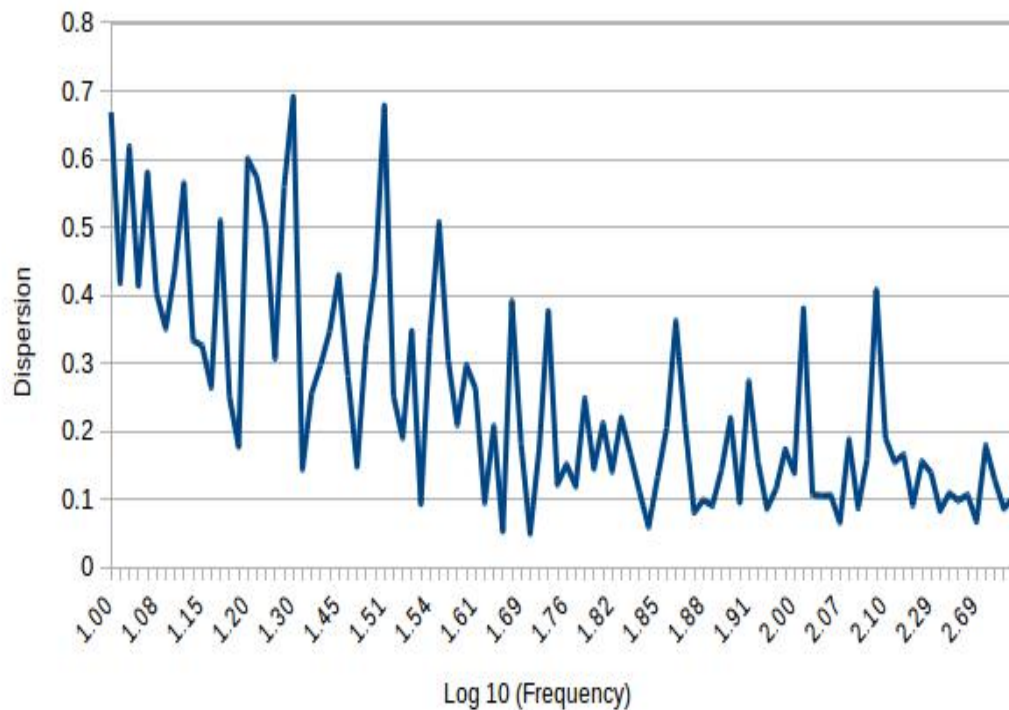


Figure 8.2: Scatter jitters of selected hundred words from different frequency bins.

and their dispersion. As shown by the non-parametric smoothers in two counts: the more frequent a word, the more uniformly conveyed it is all throughout the corpus. On the other hand, it is also clear that the relationship between frequency and DP is just probabilistic. Especially the middle frequency range contains words with very high and very low dispersions.

The dispersion measurement for all words of the corpus we have normalized the DP value to DP_{norm} , that imply under the measurement range between 0 and 1, while BNC dispersion value goes beyond the boundary and they discard these out of range values as shown in Figure 8.3. Comparison of information scattering in our corpus using the frequency profiling methodology of dispersion measure with BNC corpus are shown in Figure 8.3.

8.3.3.2 Dispersion Measure using the Standard Deviation

In addition to pure frequency profiling based dispersion measurement Rosengrens and Juilland recommends procedures to find out the dispersion measures, where they include some more information or parameters for analysis of dispersion for random variable data set such as: the standard deviation (σ) and coefficient of variation.

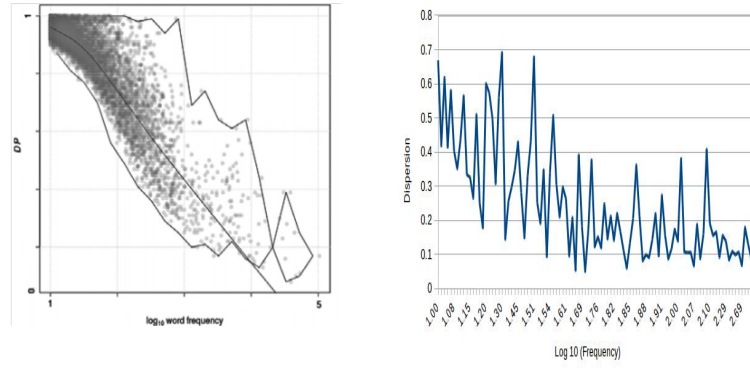


Figure 8.3: Comparison of information scattering in (a) BNC corpus words scatter behavior; (b) Scatter jitters of selected hundred words from different frequency bins.

To figure out the variations (dispersion) of the data set, relative measures of standard deviation must be computed. Coefficient of variation (C_v) or the coefficient of σ is computed below:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (f_i - \bar{f})^2}{n - 1}} \quad (8.7)$$

where $\bar{f} = \frac{f}{n}$

$$C_v = \frac{\sigma}{\bar{f}} \quad (8.8)$$

Thus, it is also defined as the proportion σ to its mean. It is given as a percentage and is utilized to compare the consistency or variability of two kinds of arrangements. For the higher C_v , variability will be higher and for the lower C_v , the consistency of the information will be higher.

Dispersion Measure by Juilland et al.:

Juilland suggests a method to find out the dispersion in data using the C_v . The value of dispersion for the data set using variations coefficient is as described below:

$$D = 1 - \frac{C_v}{\sqrt{n - 1}} \quad (8.9)$$

where $D = \text{Dispersion}$

Dispersion Measure by Rosengrens:

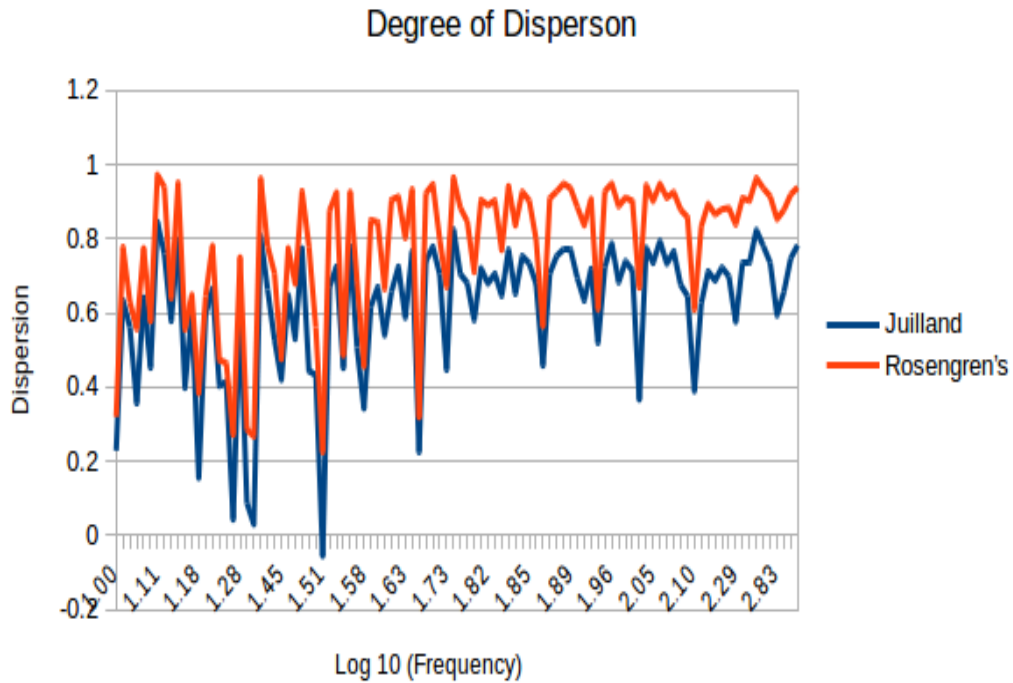


Figure 8.4: Measurement of level of information scattering for *DSHTC* Hindi corpus for the selected words Using the Juillard and Rosengrens methodology of dispersion measure.

Rosengrens suggests a method for finding the value of dispersion based on the observed frequency of corpus. Rosengrens also considered a total number of parts as denoted by n . In our Corpus, we have taken the value of n as 7 (number of data collection categories of our corpus). The resulted value of dispersion for our corpus is obtained using the below given formula:

$$D = \left(\frac{1}{n} \left(\sum_{i=1}^n \sqrt{f_i} \right)^2 \right) \times \frac{1}{f} \quad (8.10)$$

where $D = Dispersion$

$$\text{where } \min D = \frac{1}{n} = 0.17.$$

Figure 8.4 shows that the degree of dispersion for the selected words of our corpus using the Juillard and Rosengrens techniques of dispersion measures.

The results from the graph in Figure 8.4 shows that the middle frequency range contains words with very high and very low dispersions in both of the techniques.

The distribution of words is uneven for the average word frequency. The words with low-frequency also have an irregularly behavior. Compared to middle frequency words, the low frequency words are having a highly scattered behavior. The results of graph Figure 8.4 shows that dispersion of data is same in both techniques, only magnitude of the DP value is varied.

8.4 Word Frequency Distribution using Zipfs Rule

The quick decrease in frequency with few words having high frequencies is a standard measure in corpora and has been utilized as a motivation behind why huge corpora are expected to represent low frequency words precisely. So for validation the above statement Zipfs provide a rule, for the applicability of our data set as a standard corpus linguistic resource. We have conducted the Zipfs [Piantadosi (2014)] test on the data set to determine that it caters to the universality of a language principle. Zipfs rule states that, if words are arranged from the corpus in descending order of frequency (w_1, w_2, \dots, w_n) . Then, the occurrence frequency of second word w_2 is $\frac{w_1}{2}$ half times as the first word w_1 and the third word w_3 occurred roughly $\frac{w_1}{3}$ one-third as often as the first word, and so on. From this it can be concluded that, the multiplication of the rank of a word r by its frequency f , would remain constant C for every word.

$$wf_i = \frac{C}{wr_i} \quad (8.11)$$

From the above equation we can infer a speculation of this law expressing that, frequency of words decreases quickly with rank. It can also be written as:

$$wf_i = C(wr_i)^k \quad (8.12)$$

By taking log of the above equation, we get:

$$\log (wf_i) = \log C + k \log (wr_i) \quad (8.13)$$

Where $k = -1$ and C is a constant. So a $\log(f)$ and $\log(r)$ graph drawn between frequency and rank of a corpus must be linear with a slope -1 . The Figure 8.5 shows the Zipf's curve for the our corpus words. The $\log(f)$ and $\log(r)$ Zipfs graph validate that our corpus follows Zipfs rule for frequency distribution of words.

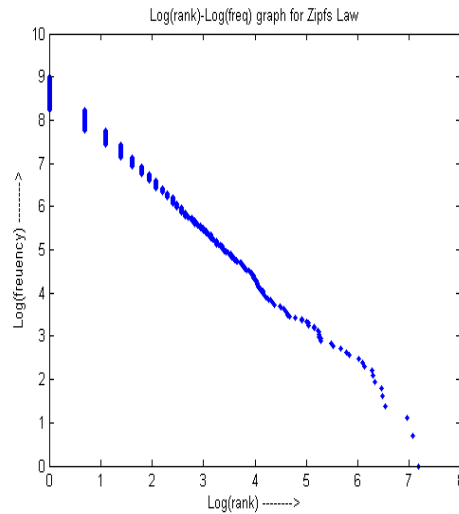


Figure 8.5: Validation of words distribution in of the our corpus through Zipfs graph.

8.5 Text Line Segmentation Results

To test the usability of our corpus for benchmarking, we have used it as the test and training set for quantitative analysis of some handwritten text segmentation algorithms.

The handwritten text segmentation algorithms described below are used for validating our corpus and available as an open source for researchers working in Devanagari Hindi handwritten text recognition.

Each technique was tested on 600 images, where, 500 images selected from the News, Science & Technology, History, Literature, Politics categories (100 forms each) and and remaining 100 images were a combination images from Architecture and Economy/Business categories. The average accuracy of segmentation algorithms is measured by Equation 8.14 as follows:

$$Accuracy = \frac{TP + TN}{(TP + TN + FP + FN)} \quad (8.14)$$

Where as:

TP = True positive.

TN = True negative.

FP = False positive.

FN = False Negative.

1. A approach presented by [S. Godara and Ahmed (2014)] is tested, to seg-

ment handwritten text document into individual text lines using smearing method. In this approach, first the input grayscale image is converted into a binary image. Then, the resultant binary image is divided into vertical stripes, where the strip size is dynamically calculated for each document based on average width of connected components. To separate foreground and background information, black blocks are placed strip-wise in the image wherever foreground pixels are present. To get a single line dis-connections present in the strips need to be filled, so dilation in horizontal direction is applied. This method was applied and tested on 600 Devanagari which were selected from our data set. The achieved average accuracy by the techniques is 90.69%.

2. The next approach tested as developed by [Deepti Khanduja and Panwar (2015)] proceeds by computing the bounding box and centroid for the 8-connected components of the document image. The achieved average accuracy by the technique is 93.28% on 600 images.
3. A Connectivity Strength Parameter (CSF) based technique is presented [Panwar and Nain (2014)]. A text-line segmentation results obtained with this technique from experimentation on 600 images of our corpus is 93.24%.

Table 8.2: Analysis of three approach used for testing our corpus for text line segmentation benchmarking.

Techniques	Number of test images	Average Accuracy
[S. Godara and Ahmed (2014)]	600	90.69%
[Deepti Khanduja and Panwar (2015)]	600	93.28%
[Panwar and Nain (2014)]	600	93.24%

Table 8.2 summarizes the complete test results. The sample result of line and word segmentation are shown in Figure 8.6 and Figure 8.7 respectively.

8.6 Summary

We have detailed here various statistical analysis techniques with the corpus information to make it more representative to describe the behavior of texts in corpus information. The statistical results presented with corpus data will make possible the construction of more satisfactory mathematical models of language. In this

अबुल फजल ने अकबर की अत्यधिक प्रशंसा की है।
 उसके ग्रंथ से तत्कालीन राजनीतिक घटनाओं के
 साथ सांस्कृतिक दशा का भी पता चलता है। उसने
 प्रत्येक महत्वपूर्ण घटना से पहले उसका प्रस्तावना
 लिखी है। उसके विवरण से तत्कालीन धार्मिक
 स्थिति पर भी प्रकाश पड़ता है। वह अकबर
 की धार्मिक उदारता से अत्यधिक प्रभावित था।

Figure 8.6: Sample result of line segmentation.

अबुल फजल ने अकबर की अत्यधिक प्रशंसा की है।
 उसके ग्रंथ से तत्कालीन राजनीतिक घटनाओं के
 साथ सांस्कृतिक दशा का भी पता चलता है। उसने
 प्रत्येक महत्वपूर्ण घटना से पहले उसका प्रस्तावना
 लिखी है। उसके विवरण से तत्कालीन धार्मिक
 स्थिति पर भी प्रकाश पड़ता है। वह अकबर
 की धार्मिक उदारता से अत्यधिक प्रभावित था।

Figure 8.7: Sample result of word segmentation.

chapter scatter behavior of the corpus to validate the quality of texts are also described.

The statistical approaches also useful to discover the major changes occurs in the theoretical landscape of linguistics. In particular, statistical approaches from corpus linguistics are frequently used to study distributional characteristics of linguistics units and reveal about cognitive processes. For example, frequency, entropy contingency, dispersion, and Zipfian distributions contribute toward exploring these notions and their roles for linguistic cognition.

Chapter 9

Conclusions and Future Work

This work presents a step towards an efficient integration of Devanagari Hindi corpus construction and, structural and linguistic annotations. In this thesis, we presented a methodology to build Devanagari Hindi corpus that is labelled with semantic roles to support the research work in the linguistic domain by proposing an effective framework. In this chapter we present the summary and conclusions drawn from our research study and feasible improvements in future to extend this work.

9.1 Thesis Summary

In chapter 2, detailed the linguists view regarding linguistic contexts and evolution of the corpus in various domain of the Natural Language Processing (NLP). The chapter explored the fundamental principles or characteristics of the corpus that should be followed during the compilation of a corpus to make it available for the long-term usability. Chapter described a detailed literature survey of the existing corpora. The chapter illustrated the current corpus-related activities in multidisciplinary research such as education, document image analysis, OCR etc.

Chapter 3 described the existing corpus supportive models and defined an annotation of the corpus with the need for annotation and standards followed by the linguists for annotation. The complete description of the corpus supportive frameworks has been carried out in three phases: corpus development, annotation of corpus and retrieval or statistical analysis of the corpus information. This chapter further explored the various levels of annotation used for different purposes. The

observation also brought in focus that most of the well-known tools are limited to machine readable text and priority toward the handwritten document annotation is very less, which is a most demanding platform for various document image analysis tasks. The functionality of tools is limited to either annotation or information retrieval.

Chapter 4 showed the compilation process of the corpus, from data selection to handwritten form generation. Here, we have explored the quantitative information of the corpus in terms of the text-page, line and word. We have also described the detailed description of the writer demographic information about geographical region, education, age, gender etc.

Chapter 5 explored the methodology and the framework that is implemented to annotate the Devanagari Hindi handwritten text image corpus in a systematically and scientifically. The implementation process of the methodology described the configuration of the backhand database configuration and encoding standards used for the annotation. Furthermore, the chapter explained the features of annotation proposal for implementation of annotation schemes to produce the GT of handwritten text as a resource. The chapter detailed the functionality and usability of the framework for the efficient and fast retrieval of stored information. The chapter showed a worked example of the GT sample image to illustrates the experimental results.

Chapter 6 is concerned with part-of-speech (POS) tagging approach for Devanagari Hindi text. The POS tagging method is designed with the help of probabilistic approach and rules-based approach. The chapter detailed the approach and rules implemented for POS tagging. An ambiguity removing module is also explained in this chapter.

Chapter 7 described chunking approach for Devanagari Hindi text and explained the rule-based model of the chunking approach. These rules are generated by detailed analysis of POS tagged corpus of Devanagari Hindi language and represented through regular expressions. Various FSM automaton are detailed for implementation of these rules.

Chapter 8 represents the evaluation of various statistical techniques with proposed framework (DSHTC) of corpus using the data-driven approach. Purpose of the statistical evaluation is to validate the corpus on different standards to fulfill the requirement of standard linguistic resource for Devanagari Hindi scripts. Corpus data has been benchmarked with both quantitative and statistical measurements.

The results of statistical analysis study are presented and compared with existing Indic scripts corpus. Outcomes also described the grammatical information about Devanagari Hindi scripts and corpus data.

9.2 Major Contributions of Research

In this work, we present the Devanagari Hindi handwritten and Unicode corpus. For the compilation and annotation of corpus we have developed a platform independent model. The framework is capable of annotating a vast database and is language independent. The framework considered both handwritten and Unicode text and provided an align transcription between them. For the Devanagari Hindi script benchmarking platforms we have provided the word level GTs, POS tagging and chunking of Devanagari Hindi handwritten documents. In addition to Devanagari Hindi corpus compilation, we have experimented on variety of statistical analysis techniques with the corpus information to make it more representatives and describing of the behavior of texts in corpus body. This work can be widely used in corpus linguistics like for benchmarking of experimental work in handwritten text recognition techniques, word substitutions, as a language tutor, as a trainer etc.

The major contributions from our research work can be summarized as follows:

1. We carried out an extensive survey on the handwritten corpus and supportive linguistic models. With the literature survey, it is concluded that there exists a sufficient number of standard databases for English, Chinese, Arabic and Japanese, while very few standard databases are available for Devanagari Hindi. Compared to these languages very poor resources are available for Devanagari script Hindi.
2. We have considered both handwritten and aligned Unicode text in our method and developed structural mark-up and linguistic annotated handwritten Devanagari Hindi corpus. The overall corpus consists of 1650 handwritten text forms, filled by 1650 writers from different age groups and scholastic qualifications with geographically diversion regions.
 - The database contains 1650 handwritten text form, filled by 1650 writers from different age groups, with different educational qualifications and geographical location.

- Each filled handwritten text form contains 4 to 5 line of text. There are 7425 Devanagari Hindi handwritten text-lines.
- Each filled handwritten text form contains 70 to 80 words. In addition of this the database contains 5775 Hindi printed text lines. There are a total of 123,750 words in the data set.

However, the compilation of corpus contains various amount of significant information such as printed texts, handwritten texts, digits etc. which make it useful to fulfil the different requirements of research.

3. Most of the corpora related work in literature are limited with corpus development instead of annotation and statistical validations of the corpus information. We identified the need for implementing a corpus framework which can provide a systematic structural annotation of the raw corpus as a standard linguistic resource, and provided it for the Devanagari Hindi script. The corpus structure is platform independent and could be scaled to many scripts.
4. For systematic integration and indexing of the corpus information, we have designed a framework, which provides an aligned transcription between handwritten and Unicode texts. The framework generates an auto-indexing number at each level (text-page, line and words). We also extend the functionality and usability of the corpus, by adding the grammatical information (tags, synonyms, antonyms etc.) with the plain text so that, it could be used as a linguistic resource.
5. Existing models available for handwritten documents annotation are limited in their tagging levels. They usually provide only the text-page annotations. GTLC, a model developed for the annotation of Chinese handwritten documents, extends this functionality up to lines and words. We provide automatic textual region GT annotations at different level for text-pages, lines and words for Devanagari Hindi script, which is untouched in Devanagari Hindi handwritten GT field. Furthermore, to ascertain the applicability of the corpus as a standard platform for OCR training and benchmarking, we have experimented three text-line segmentation techniques to validate the use of GT annotation.
6. The statistical analysis is one of the best ways for profiling the corpus information. From the literature, it was noted that there exist only two models

with the statistical analysis, that is, the Matrix and the WordSmith; however, none of the tools combined the capability of annotation awareness with the statistical analysis. We have done both the annotations and statistical analysis.

7. We have presented an automatic POS approach for Devanagari Hindi text, designed with the help of probabilistic approach and rules-based approach. We also presented a chunking approach for Devanagari Hindi text. The chunking method does not use any manual chunked corpus of Hindi language. It uses only the POS information of the text as contextual information. The approach is based on rule-based model. These rules are generated by detailed analysis of POS tagged corpus of Devanagari Hindi language and represented through regular expressions.
8. We have focused on both the quantitative and statistical analysis to validate the corpus as a standard linguistic resource.
 - (a) The quantitative measurements of the corpus have been done to explore the contents of the corpus regarding Equality, Representation, Quantity etc.
 - (b) We have also done a statistical analysis on the compiled corpus data to discover the behavior of corpus information in terms of Quality, Homogeneity, Balancing, Consistency etc. to validate the purpose of data collection (corpus compilation) and ascertain the goodness of model. In the statistical analysis, first we have applied the Entropy and Perplexity measurements. The Entropy is used to explore the information associated with the corpus data and number of bits required for encoding the corpus and perplexity is a mathematical evaluation to describe the information encapsulated in the corpus for generating mathematical aspects about the language. Further, the results are compared with other Indic scripts corpora for relative comparison and to discover the similarity and dissimilarity.
 - (c) Furthermore in statistical analysis, the dispersion measurement techniques are applied to discover the scramble behavior of corpus information in linguistic expression. Results prove that the dispersion of the corpus information reflects the interpretations of overall distribution in the corpus in a significant way. We have also applied a Zipfs test on the corpus to prove the hypothesis test of the linguistic standard.

9. For easy representation and interpretation, we have provided an effective concept of data configuration and a web-based interface for information extraction from a handwritten document, which is a robust and unique approach.

9.3 Limitations and Future Scope

[Bhattachara and Chaudhuri (2009)] developed a mixed numerals handwritten databases of Devanagari Hindi and Bangla, which includes only isolated handwritten numeral samples. [Dongre and H.Mankar (2012)] developed database for Devanagari Hindi characters only. [Bhattacharya and Chaudhuri (2005)] developed Indian scripts digit databases for Bangla, Oriya and Devanagari Hindi is again limited to collection only digit. [Jayadevan (2011)] developed a Devanagari legal amount word handwritten database of Devanagari Hindi and Marathi. These handwritten corpus or data set on Devanagari script Hindi are mainly of numerals and characters and does not provide the sentences, annotations, GT, and also does not include the transcription of handwritten corpus.

1. To increase the scale of the proposed framework, for example to include format free handwritten text pages.
2. To extend the corpus utility by adding bidirectional aligned translation between Devanagari Hindi and other Indic scripts.
3. To perform cross corpus comparison between Devanagari Hindi and other similar Indic scripts like frequency and statistical analysis.

Appendix A

List of Publications

International Journals

1. Maninder Singh Nehra, Deepa Modi, Neeta Nain and Mushtaq Ahmed. Part-of-speech Tagging for Hindi Corpus in Poor Resource Scenario. *Journal of Multimedia Information System (JMIS)*, Vol. 5(3). pp. 147-154. 2018. (Published)
2. Maninder Singh Nehra, Deepa Modi, Neeta Nain and Mushtaq Ahmed. A Rule-based Chunker for Hindi Language. *ACM Transaction on Asian and Low-Resource Language Information Processing (TALLIP)*, 2017. (In process)
3. Maninder Singh Nehra, Neeta Nain, Mushtaq Ahmed and Prakash Chodhary. Handwritten Text Image Corpus for Multidisciplinary Research on Devnagari Script Hindi: Benchmarking and Annotation. *ACM Transaction on Asian and Low-Resource Language Information Processing*, 2018. (Communicated)

International Conferences

1. Maninder Singh Nehra, Neeta Nain and Mushtaq Ahmed. POS Tagging and Structural Annotation of Handwritten Text Image Corpus of Devnagari Script. In *ICETCE-2019*, Springer, Jaipur, India, 2019. (Published)

2. Maninder Singh Nehra, Neeta Nain, Mushtaq Ahmed, Prakash Chodhary and Deepa Modi. Amalgamated Approach for Devnagari Script Corpus for OCR Demographic Purpose and XML for Linguistic Annotation. In *Signal-Image Technology & Internet-Based Systems (SITIS), 2017 13th International Conference*, IEEE, Jaipur, India, 2017. (Published)
3. Maninder Singh Nehra, Neeta Nain and Mushtaq Ahmed. An Annotated Hindi Corpus of Handwritten Text Image and Benchmarking of Corpus. In *IAPR Summer School on Document Analysis: Document Informatics, 23rd-28th January 2017*, Jaipur, India. (Published)
4. Maninder Singh Nehra, Neeta Nain and Mushtaq Ahmed. Novel Database for Hindi Handwritten Text Recognition and Comparative Survey of Handwritten Databases. In *Springer 24'ICFIDCAA-2016, 22-24 August 2016*, Anand International College of Engineering, Jaipur, India. (Published)
5. Maninder Singh Nehra, Neeta Nain and Mushtaq Ahmed. Benchmarking of Text Segmentation in Devnagari Handwritten Document. In *IEEE PIICON-2016, Under the agis of IIT Delhi chapter, GEC Bikaner, Rajasthan, India, 2016*. (Published)
6. Anushika Jain, Maninder Singh Nehra, Neeta Nain and Mushtaq Ahmed. An Accelerated Approach for Generalized Entropy Based MRI Image Segmentation. In *Springer ICACDS-2016, 11-12 November 2016*, Ghaziabad (UP,) India. (Published)
7. Maninder Singh Nehra, Prakash Chodhary and Neeta Nain. A Framework for compilation of Multi- Lingual Handwritten Databases: Four Levels Ground-Truth. In *The 11th International Conference on Signal Image Technology and Internet Based Systems-2015*, IEEE, Bangkok, 2015. (Published)
8. Maninder Singh Nehra, Neeta Nain and Mushtaq Ahmed. Handwritten Devnagari Script Database Development for off-line Hindi Character with Matra (modifiers). In *International Conference on Recent Cognizance in Wireless Communication & Image Processing (ICRWIP)*, Springer, Jaipur, India, 2015. (Published)
9. Maninder Singh, Deepa Modi, Neeta Nain, Mushtaq Ahmed. Survey of Techniques for Two Level Corpus Annotation for Hindi. In *NCMAC, MNIT, Jaipur, India, 2015*. (Published)

Bibliography

Technology development for indian languages.

(1995). Isi bengali corpus.

(2007). The british national corpus: Bnc xml edition). <http://www.natcorp.ox.ac.uk/>. Distributed by Oxford University Computing Services on behalf of the BNC Consortium.

Agns Tutin, Meriam Haddara, R. M. and Orasan, C. (2004). Annotation of anaphoric expressions in an aligned bilingual corpus. In *Proceedings of the Fourth Language Resources and Evaluation Conference (LREC)*, pages 267–270.

Agrawal SS., Mandal Abhimanue, B. S. and Minakshi, M. (2014). Statistical analysis of multilingual text corpus and development of language models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 26–31.

Akshar Bharati, Dipti Misra Sharma, L. B. R. S. (2006). *AnnCorra : Annotating Corpora*. Language Technologies Research Centre, IIIT, Hyderabad.

Akshay, S. B. and Sangal, R. Hmm based chunker for hindi. pages 126–131.

ALAEI, A., PAL, U., and NAGABHUSHAN, P. (2012). Dataset and ground truth for handwritten text in four different scripts. *International Journal of Pattern Recognition and Artificial Intelligence*, 26(04):1253001.

Ali, W. and Hussain, S. (2010). A hybrid approach to urdu verb phrase chunking. *Proceedings of the 8th Workshop on Asian Language Resources*, pages 136–142.

Aniket Dalal, Kumar Nagaraj, U. S. and Shelke, S. (2006). Hindi part-of-speech tagging and chunking : A maximum entropy approach. *IIT Bombay CSE*.

Anthony, L. (2004). Antconc: A learner and classroom friendly, multi-platform corpus analysis toolkit. In *IWLeL 2004: An Interactive Workshop on Language e-Learning*, pages 7–13.

-
- Antony (2011a). *Parts Of Speech Tagging for Indian Languages: A Literature Survey*. International Journal of Computer Applications (0975 6 8887), India.
- Antony, P. (2011b). Parts of speech tagging for indian languages : A literature survey.
- Ashish and Sinha, A. A new approach for hmm based chunking for hindi. *Term Project on Speech and Natural Language Processing*.
- Asopa, S., Asopa, P., Mathur, I., and Joshi, N. (2016). Rule based chunker for Hindi. *Proceedings of the 2016 2nd International Conference on Contemporary Computing and Informatics, IC3I 2016*, pages 442–445.
- Avinesh and G, K. Part-of-speech tagging and chunking using conditional random fields and transformation based learning.
- Baker, Fellbaum, F. P. (2008). Masc: The manually annotated sub-corpus of american english.
- Baker, P., H. A. M. T. C. H. . G.-R. (2002). Emille, a 67million word corpus of indic languages: Data collection, mark-up and harmonisation.
- Baskaran, S. (1998). Hindi POS Tagging and Chunking. *Microsoft Research India Bangalore, INDIA*.
- Bharati Akshar, Rao K Prakash, S. R. and S.M., B. (2002). Basic statistical analysis of corpus and cross comparison among corpora. In *Proceedings of 2002 International Conference on Natural Language Processing*, India.
- Bhattachara, U. and Chaudhuri, B. B. (2009). Handwritten numeral database of indian scripts and multistage recognition of mixed numerals. In *IEEE trans. on Pattern Analysis and Machine Intelligence*, pages 444–457.
- Bhattacharya, U. and Chaudhuri, B. B. (2005). Databases for research on recognition of handwritten characters of indian scripts. In *Proc. 8th ICDAR*, pages 789–793.
- Bindu.M.S and Idicula, S. M. (October 2011). A hybrid model for phrase chunking employing artificial immunity system and rule based methods. *International Journal of Artificial Intelligence Applications (IJAIA)*, Vol.2(No.4).
- Bird, S. (2009). *Natural Language Processing with Python*. O Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472., United States of America.

-
- BNC (2003). The british national corpus:bnc world). <http://www.natcorp.ox.ac.uk/>. Distributed by Oxford University Computing Services on behalf of the BNC Consortium.
- BNC (2007). The british national corpus: Bnc xml edition). <http://www.natcorp.ox.ac.uk/>. Distributed by Oxford University Computing Services on behalf of the BNC Consortium.
- Bonaventura, P., Howarth, P., and Menzel, W. (2000). Phonetic annotation of a non-native speech corpus.
- Bworld (2015). http://commons.org/wiki/file:states_of_south_asia.png. *Online; accessed 19–July – 2015*.
- Carletta J., Evert S., H. U. K. J. (2005). The nite xml toolkit: data model and query. *Language Resources and Evaluation Journal (LREJ)*, 39(4):313–334.
- Chandan Mittal, V. G. and Singh, U. (December 2015). Hmm chunker for punjabi. *Indian Journal of Science and Technology*, Vol 8(35),.
- Chaudhuri, B. A complete handwritten numeral database of bangla 6 a major indic script. *CVPR Unit, Indian Statistical Institute, Kolkata-108, India*.
- Chris Brew, David McKelvie, R. T. H. T. A. M. The xml library lt xml. <https://www.ltg.ed.ac.uk/software/ltxml/>.
- Christophe Laprun, Jonathan G. Fiscus, J. G. and Pajot, S. (2002). A practical introduction to atlas. In *Proc. of Int. Conf. on Language Resources and Evaluation*, Las Palmas, Spain.
- CLARIN (2009). Standards for text encoding: A clarin shortguide. <http://www.clarin.eu/documents>. Distributed by Oxford University Computing Services on behalf of the BNC Consortium.
- Collins, H. (1995). Collins cobuild english dictionary (2nd edition).
- Cunningham, H., Humphreys, K., Gaizauskas, R., and Wilks, Y. (1997). Gate: A general architecture for text engineering. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, pages 29–30, Washington, DC, USA. Association for Computational Linguistics.
- D. Llorens, F. Prat, A. M. J. M. V. (2011). The ujipenchars database: A pen-based database of isolated handwritten characters. pages 2647–2651.

-
- Da-Han Wang, Cheng-Lin Liu, J.-L. Y. X.-D. Z. (2009). Casia-olhwdb1: A database of online handwritten chinese characters. *In Proc. of ICDAC IEEE*, pages 1206–1210.
- Dalal, A., Kumar, N., Uma, S., Sandeep, S., and Pushpak, B. (2007). Building Feature Rich POS Tagger for Morphologically Rich Languages : Experiences in Hindi. *5th International Conference on Natural Language Processing*, page 9.
- Dandapat, S. (2009a). Part-of-Speech Tagging for Bengali. pages 1–132.
- Dandapat, S. (2009b). Part-of-Speech Tagging for Bengali. pages 1–132.
- David and Alicia (2014). Bh2m: the barcelona historical handwritten marriages database. *In ICPR of IEEE*, pages 256–261.
- Davies, M. (2005). A frequency dictionary of spanish.
- Deepa Modi, Maninder Singh Nehra, N. N. and Ahmed, M. (2015). A survey of techniques for two level corpus annotation for hindi. *International Bulletin of Mathematical Research Volume XX, Issue X, 2015*, pages 1–10.
- Deepti Khanduja, N. N. and Panwar, S. (15(1):2:162:10, November 2015). A hybrid feature extraction algorithm for devanagari script.
- Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, pages 141–142.
- Dipanjan Das, Monojit Choudhury, S. S. and Basu, A. (2005). An affinity based greedy approach towards chunking for indian languages.
- Dongre, V. J. and H.Mankar, V. (2012). Development of comprehensive devnagari numeral and character database for offline handwritten character recognition. *In Proc. Applied Computational Intelligence and Soft Computing*, pages 1–5.
- Douglas Biber, S. C. and Reppen, R. (1998). Corpus linguistics: Investigating language structure and use.
- E.Kavallieratou (2001). The gruhd database of greek unconstrained handwriting. *In International Conf. of EEE*, pages 561–565.
- Ekbal, A., Haque, R., and Bandyopadhyay, S. (2008). Maximum Entropy Based Bengali Part of Speech Tagging. *A. Gelbukh (Ed.), Advances in*, pages 67–78.

-
- Elliman, D. and Sherkat, N. (2001). A truthing tool for generating a database of cursive words. In *Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on*, pages 1255–1262.
- Elmasri, R. and Navathe, S. B. (2000). Fundamentals of database systems.
- Evert, S. (Published in 2005). The statistics of word cooccurrences: Word pairs and collocations.
- Evert, S. and Team, T. O. D. (2010). The ims open corpus workbench (cwb): Cqp query language tutorial. http://cwb.sourceforge.net/files/CQP_Tutorial/node1.html.
- Farooqui, A. S. (February 2016). corpus-based study of academic-collocation use and patterns in postgraduate computer science students? writing. *University of Essex*.
- Farrah, S., Manssouri, H. E., Ziyati, E. H., and Ouzzif, M. (2018). An Hybrid Approach to Improve Part of Speech Tagging System.
- Fischer, A., Indermühle, E., Bunke, H., Viehhauser, G., and Stolz, M. (2010). Ground truth creation for handwriting recognition in historical documents. In *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems, DAS '10*, pages 3–10, New York, NY, USA. ACM.
- Francis (1992). Language corpora b. c. in svartvik, j. (ed.). In *Directions in Corpus Linguistics: Proceedings of Nobel Symposium 82, Stockholm,*, pages 17–32, Mouton de Gruyter, Berlin.
- Fu, G.-H., Xu, R.-F., Luke, K.-K., and Lu, Q. (2005). Chinese text chunking using lexicalized hmms. In *2005 International Conference on Machine Learning and Cybernetics*, volume 1, pages 7–12 Vol. 1.
- G., A. and L., B. (1998). The bnc handbook: Exploring the british national corpus with sara. Edinburgh. Edinburgh University Press.
- Ganesan, M. (2003). Tamil corpus generation and text analysis.
- Gao, H., Huang, D., Yang, Y., and Li, L. (2006). Chinese chunking using esvm-knn. In *2006 International Conference on Computational Intelligence and Security*, volume 1, pages 731–734.
- Garg, N., Goyal, V., and Preet, S. (2012). Rule Based Hindi Part of Speech Tagger. 2(December):163–174.

-
- Gatos, B., Louloudis, G., Causer, T., Grint, K., Romero, V., Sanchez, J., Toselli, A., and Vidal, E. (2014). Ground-truth production in the transcriptorium project. In *Document Analysis Systems (DAS), 2014 11th IAPR International Workshop on*, pages 237–241.
- Geoffrey Leech, Stig Johansson, R. G. and Hoffland, K. (1986). The lob corpus: Pos-tagged version.
- Goyal and Lalit (2011). Comparative analysis of printed hindi and punjabi text based on statistical parameters. pages 209–213.
- Grice, M., G. M. L. G. W.-M. and Wilson, A. (2000). Representation and annotation of dialogue. In *Handbook of multimodal and spoken dialogue systems: Resources, terminology and product evaluation*, eds. D. Gibbon, I. Mertins and R. K. Moore, pages 1–101.
- Gries, S. T. (2008). Measures of dispersion in corpus data: a critical review and a suggestion.
- Grosicki, E. (2009). Icdar 2009 handwriting recognition competition. *IN proc. of ICDAR*, pages 1398–1402.
- Grnnum, N. (2009). A danish phonetically annotated spontaneous speech corpus (danpass). *Speech Communication*, 51(7):594 – 603. Research Challenges in Speech Technology: A Special Issue in Honour of Rolf Carlson and Bjrn Granstrm.
- Gunasekara, D., Welgama, W. V., and Weerasinghe, A. R. (2017). Hybrid Part of Speech tagger for Sinhala Language. *16th International Conference on Advances in ICT for Emerging Regions, ICTer 2016 - Conference Proceedings*, pages 41–48.
- Gupta, V., Joshi, N., and Mathur, I. (2016). CRF based Part of Speech Tagger for domain specific Hindi Corpus. pages 14–18.
- Hasan, F. M. (2006). Comparison of different pos tagging techniques for some south asian language.
- Hillsdale, N. (1995). The chldes project: tools for analyzing talk.
- Hossein (2007). Introducing a very large dataset of handwritten farsi digits and a study on their varieties. In *Elsevier Pattern Recognition Letters*, pages 11336–1141.
- Hull, J. J. (1994). A database for handwritten text recognition research. In *IEEE Transaction on Pattern Analysis and Machine Intelligence*, pages 550–554.

-
- Jain, U. and Kaur, J. (10 Oct 2015). Text chunker for punjabi. *International Journal of Current Engineering and Technology*, Vol.5(No.5).
- Janet Holmes, Bernadette Vine, G. J. B. V. (1998). The wellington corpus of spoken new zealand english.
- Jayadevan, R. (2011). Database development and recognition of handwritten devanagari legal amount words. In *ICDAR Proc.*, pages 304–307.
- Johansson, S. and Hansen, C. F. (2008). The oslo multilingual corpus (1999-2008). In the Faculty of Humanities, U. o. O., editor, *The Oslo Multilingual Corpus is a product of the interdisciplinary research project Languages in Contrast (SPRIK)*.
- Jones, R. and Tschirner, E. (2005). A frequency dictionary of german.
- Joshi, N., Darbari, H., and Mathur, I. (2013). HMM based POS tagger for Hindi. *Proceedings of 2nd International Conference on Artificial Intelligence and Soft Computing*, pages 341–349.
- Juilland A.G., B. D. R. and C., D. (1970). Frequency dictionary of french words.
- Kazuaki Maeda, Steven Bird, X. M. and Lee, H. (2002). Creating annotation tools with the annotation graph toolkit. In *Proc. 3rd Int. Conf. on Language Resources and Evaluation*, pages 1914–1921.
- Kiani, S., Akhavan, T., and Shamsfard, M. (2009). Developing a persian chunker using a hybrid approach. In *2009 International Multiconference on Computer Science and Information Technology*, pages 227–234.
- Kim and Park (1993). Handwritten korean character image database pe92. In *In Proc. of IEEE*, pages 470–473.
- Kiril Simov, Alexander Simov, K. I. I. G. H. G. (2003). The clark system tools xml based corpora development. In *In: Workshop on Balkan Language Resources and Tools*, pages 235–238, Thessaloniki, Greece.
- Kishorjit Nongmeikapam, Chiranjiv Chingangbam, N. K. B. V. and Bandopadhyay, S. (June 2014). Chunking in manipuri using crf. *International Journal on Natural Language Computing (IJNLC)*, Vol. 3(No.3):436–438.
- Krajka, J. (2009). Handbook of research on web 2.0 and second language learning. *Warsaw School of Social Psychology, Poland*, page 21.

-
- Lafferty (2001). *Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data*. ACM, University of Pennsylvania ScholarlyCommons.
- Lata, S. (2011). Challenges of multilingual web in india : Technology development & standardization perspective. *International Bulletin of Mathematical Research*, pages 1–63.
- Lawgali, A. and Bouridane (2013). Hacdb: Handwritten arabic characters database for automatic character recognition. In *Visual Information Processing (EUVIP), 2013 4th European Workshop on*, pages 255–259.
- Lehmann, H. M., S. P. and Hoffmann, S. (2000). Bncweb. in kirk, j. m. (ed.) corpora galore. pages 259–266, Rodopi, Amsterdam.
- Long (2011). Mrg-ohtc database for online handwritten tibetan character recognition. In *ICDAR*, pages 207–211.
- Lonsdale and Bras (2009). A frequency dictionary of french.
- Majumder, Z. and Khan (2006). Analysis and observations from a bangla news corpus.
- Majumder KMYA, Z. I. and M., K. (2006). Analysis and observations from a bangla news corpus. In *International Conference on Computer and Information Technology (ICCIT 2006)*.
- Malik (2009). A new large urdu database for off-line handwriting recognition. In *Inter. Conf.*, pages 1–9.
- Marcus, M., S. B. and Marcinkiewicz, M. (1993). Building a large annotated corpus of english: the penn treebank. *Journal of Computational Linguistics*, 19(2):313–330.
- Marie Garnier, A. R. and Saint-Dizier, P. (2009). Correcting errors using the framework of argumentation: Towards generating argumentative correction propositions from error annotation schemas. In *23rd Pacific Asia Conference on Language, Information and Computation*, pages 140–149.
- Martha Palmer, P. K. and Gildea, D. (2005). The proposition bank: An annotated corpus of semantic roles. *Journal of Computational Linguistics*, 31(1):71–106.
- Marti, U.-V. and Bunke, H. (2002). The iam-database: an english sentence database for offline handwriting recognition. *International Journal on Document Analysis and Recognition*, 5(1):39–46.

-
- McEnery and Wilson (2001). *Corpus linguistic*.
- McEnery, T. and Wilson, A. (1996). *Corpus linguistics*. In *Edinburgh University Press*, Edinburgh.
- MEYER, C. F. (2002). *English corpus linguistics an introduction*.
- Mishra, N. and Mishra, A. (2011). Part of speech tagging for hindi corpus. *Proceedings - 2011 International Conference on Communication Systems and Network Technologies, CSNT 2011*, pages 554–558.
- Modi, D. and Nain, N. (2016). Part-of-Speech Tagging of Hindi Corpus Using Rule-Based Method. (January).
- Mohnot, K., Bansal, N., Singh, S. P., and Kumar, A. (2014). Hybrid approach for Part of Speech Tagger for Hindi language. 4(1):25–30.
- Muljono, U. A. and Supriyanto, C. (2017). Morphology Analysis for Hidden Markov Model based Indonesian Part-of-Speech Tagger. (0):237–240.
- Murthy, K. N. and Badugu, S. (2013). A New Approach to Tagging in Indian Languages. *Research in Computing Science*, 70:45–56.
- Nakagawa, M. (2004). Collection of on-line handwritten japanese character pattern databases and their analyses. *In IJDAR*, pages 69–81.
- Narayan, R., Chakraverty, S., and Singh, V. P. (2014). Neural network based parts of speech tagger for Hindi. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 3(PART 1):519–524.
- Nawwaf (1999). A new comprehensive database of handwritten arabic words, numbers, and signatures used for ocr testing. *In Proceedings of the 1999 IEEE Canadian Conference on Electrical and Computer Engineering*, pages 706–768.
- Nemeth and Zainko (2002). Multilingual statistical text analysis, zipfs law and hungarian speech generation. *Acta Linguistica Hungarica*, (34):385–401.
- Nicoletta Calzolari, A. Z. (1996). The eagles/isle initiative for setting standards: the computational lexicon working group for multilingual lexicons. *Language Resources and Evaluation Journal (LREJ)*, pages 1–18.
- Njah, S. (2012). Mayastroun- a multilanguage handwriting database. *In ICFHR*, pages 308–312.

-
- Ojha, A. K., Behera, P., Singh, S., and Jha, G. N. (2015). Training & Evaluation of POS Taggers in Indo- Aryan Languages : A Case of Hindi , Odia and Bhojpuri. (November 2015):524–529.
- Padma, M. C. and Prathibha, R. J. (2016). MORPHEME BASED PARTS OF SPEECH TAGGER FOR KANNADA. (7):202–206.
- Pal, U. (2012). Dataset and ground truth for handwritten text in four different scripts. *In Int. Journal of Pattern Recognition and AI*.
- Pandian, S. L. and Geetha, T. V. (2008). Morpheme based Language Model for Tamil Part-of-Speech Tagging. *Technology*.
- Panwar, S. and Nain, N. (60(6):4326439, 2014). A novel segmentation methodology for cursive handwritten documents.
- Patel, C. and Gali, K. (2008a). Part-Of-Speech Tagging for {G}ujarati Using Conditional Random Fields. *Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages*, (January):117–122.
- Patel, C. and Gali, K. (2008b). *Part-Of-Speech Tagging for Gujarati Using Conditional Random Fields*. Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages, IIIT, Hyderabad.
- Patil, N. V. (2018). POS Tagging for Marathi Language using Hidden Markov Model. (1):409–412.
- Pattabhi Rao, Vijay Sunder, V. and Sobha (15 May 2015). A text chunker and hybrid pos tagger for indian languages.
- Paul, R. (2003). Matrix: A statistical method and software tool for linguistic analysis through corpus comparison. *In PhD thesis, Computing Department, Lancaster University*.
- Peb Ruswono Aryan, Iping Supriana, A. P. (2011). Development of indonesian handwritten text database offline character recognition. *In International Conference Electrical Engineering and Informatics*, pages 1–5.
- Pechwitz, M. (2003). Hmm based approach for handwritten arabic word recognition using the ifn/enit6database. *In ICDAR*, pages 1–5.
- Piantadosi, S. (2014). Zipf’s word frequency law in natural language: A critical review and future directions. *Psychonomic Bulletin Review*, 21(5):1112–1130.

-
- Pradipta Ranjan, Harish V. Sudeshna, S. A. B. (1999). Part of speech tagging and local word grouping techniques for natural language parsing in hindi. *Communication Empowerment Laboratory, IIT Kharagpur*.
- Przepirkowski, A. and Banski, P. (2011). Which xml standards for multilevel corpus annotation? 6562:400–411.
- Quirk, R. (1960). Towards a description of english usage. *Transactions of the Philological Society*, doi: 10.1111/j.1467-968X.1960.tb00308.x.
- Ram, V. S. and Lalitha, S. (2004). Noun phrase chunker using finite state automata for an agglutinative language. *AU-KBC Research Centre*, pages 218–224.
- Raza, A. (2012). Benchmark urdu handwritten sentence database with automatic line segmentation. *In ICFHR*, pages 491–496.
- Rekha and Reghu (19-21 November 2015, Trivandrum). Text chunker for malayalam using memory-based learning. In *2015 International Conference on Control, Communication Computing India (ICCC)*.
- Renouf (1987). Corpus development. pages 1–40.
- Rinku, T. S., Rajan, M., and Bhojane, V. (2014). Various Approaches Used for Tagging and Chunking in Malayalam. 5(5):1062–1066.
- Rishikesh (2018). Parts Of Speech Tagger for Maithili Language Using HMM. 7(4):206–211.
- Rizvi, A. (2010). ID3 Algorithm .
- Romary (2003). International standard for a linguistic annotation framework. *Proceedings of HLT-NAACL03 Workshop on the Software Engineering and Architecture of Language Technology*, pages 25–30.
- Romary (2004). A registry of standard data categories for linguistic annotation. *Proceedings of the Fourth International Language Resources and Evaluation Conference (LREC)*, pages 135–139.
- Rosengren (1971). The quantitative concept of language and its relation to the structure of frequency dictionaries. In *tudes de linguistique applique (Nouvelle Serie)*, pages 3–27.

-
- Ruslan Mitkov, Richard Evans, C. O. C. B. L. J. and Sotirova, V. (2004). Coreference and anaphora: developing annotating tools, annotated resources and annotation strategies. In *Proceedings of the Fourth Language Resources and Evaluation Conference (LREC)*, pages 265–274.
- Russell, B., Torralba, A., Murphy, K., and Freeman, W. (2008). Labelme: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1-3):157–173.
- S. Godara, N. N. and Ahmed, M. (2014). Handwritten urdu script segmentation using hybrid approach.
- Sabri (2012). Khatt: Arabic offline handwritten text database. In *ICFHR*, pages 449–454.
- Sachin Pawar, N. R. and Bhattacharyya, P. (Aug 2015). Noun phrase chunking for marathi using distant supervision.
- Saharia, N., Das, D., Sharma, U., and Kalita, J. (2009). Part of Speech Tagger for Assamese Text. *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, (January):33–36.
- Saito T, Yamada H, Y. K. (1985). On the data base etl9 of handprinted characters in jis chinese characters and its analysis (in japanese). *Trans IECE Jpn J68-D(4)*, pages 757–764.
- Sankar, Arnab Dhar, S. B. and Garain, U. (2011). On development and evaluation of a chunker for bangla. In *2011 Second International Conference on Emerging Applications of Information Technology*, pages 321–324.
- Sarkar, R. (2012). Cmaterdb1: a database of unconstrained handwritten bangla and banglaenglish mixed script document image. In *IJDAR Springer*, pages 1–5.
- Sarkar, R. and Basu (2012). Cmaterdb1: a database of unconstrained handwritten bangla and banglaenglish mixed script document image. In *Proc. of IJDAR*, pages 71–83.
- Sarkar, R., Das, N., Basu, S., Kundu, M., Nasipuri, M., and Basu, D. (2012). Cmaterdb1: A database of unconstrained handwritten bangla and bangla-english mixed script document image. *Int. J. Doc. Anal. Recognit.*, 15(1):71–83.

-
- Saund, E., Lin, J., and Sarkar, P. (2009). Pixlabeler: User interface for pixel-level labeling of elements in document images. In *Proceedings of the 2009 10th International Conference on Document Analysis and Recognition, ICDAR '09*, pages 646–650, Washington, DC, USA. IEEE Computer Society.
- Scott, M. (2001). Comparing corpora and identifying key words, collocations, and frequency distributions through the wordsmith tools suite of computer programs. In Ghadessy, M., H. A. and Roseberry, R. L. e., editors, *Small corpus studies and ELT: theory and practice*, pages 47–67. John Benjamins, Amsterdam.
- Sharma, S. K. and Lehal, G. S. (2011). Using Hidden Markov Model to improve the accuracy of Punjabi POS tagger. *2011 IEEE International Conference on Computer Science and Automation Engineering*, 2:697–701.
- Shrivastava, M. and Bhattacharyya, P. (2008). Hindi POS Tagger Using Naive Stemming : Harnessing Morphological Information Without Extensive Linguistic Knowledge. *6th International Conference on Natural Language Processing*.
- Siddiq, S., Hussain, S., Ali, A., Malik, K., and Ali, W. (2010). Urdu noun phrase chunking - hybrid approach. In *2010 International Conference on Asian Language Processing*, pages 69–72.
- Sinclair, J. (2004). Intuition and annotation - the discussion continues, in advances in corpus linguistics. *Papers from the 23rd International Conference on English Language Research on Computerized corpora (ICAME 23)*, pages 39–59.
- Singh, J., Joshi, N., and Mathur, I. (2013). Development of Marathi part of speech tagger using statistical approach. *Proceedings of the 2013 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2013*, pages 1554–1559.
- Singh, S., Gupta, K., Shrivastava, M., and Bhattacharyya, P. (2006). Morphological richness offsets resource demand-experiences in constructing a POS tagger for Hindi. *Proceedings of the COLING/ACL on Main conference poster sessions, COLING-ACL '06*, (July):779–786.
- Singh, S., Jha, G., and Nath (2008). Demo : Part-of-speech tagger for bhojpuri. pages 2–3.
- Sivaji Bandyopadhyay, A. E. and Halder, D. (December 2006). Hmm based pos tagger and rule-based chunker for bengali.

-
- Slimane, F., Ingold, R., Kanoun, S., Alimi, A., and Hennebert, J. (2009). A new arabic printed text image database and evaluation protocols. In *Document Analysis and Recognition, 2009. ICDAR '09. 10th International Conference on*, pages 946–950.
- Slimane1, F. (2014). Icfhr2014 competition on arabic writer identification using ahtid/mw and khatt databases. In *ICFHR*, pages 797–802.
- Smriti Singh, O. P. D. and Sarma, V. M. (December 2012.). Noun group and verb group identification for hindi. In *Proceedings of COLING 2012 , Mumbai*, pages 2491–2505.
- Somaya (2004). A data base for arabic handwritten text recognition research. In *The International Arab Journal of Information Technology*, pages 117–121.
- Sorrentino, S., Bergamaschi, S., and Parmiggiani, E. (2012). A supervised method for lexical annotation of schema labels based on wikipedia. In Atzeni, P., Cheung, D., and Ram, S., editors, *Conceptual Modeling*, volume 7532 of *Lecture Notes in Computer Science*, pages 359–368. Springer Berlin Heidelberg.
- Sthrenberg, M. (2012). The tei and current standards for structuring linguistic data. *Journal of the Text Encoding Initiative*, 3:1–14.
- Su (2013). Chinese handwriting database named hit-mw, designed to facilitate chinese handwritten text recognition. In *SpringerBriefs in Electrical and Computer Engineering*, pages 1–27.
- Su, T., Zhang, T., and Guan, D. (2007). Corpus-based hit-mw database for offline recognition of general-purpose chinese handwritten text. *International Journal of Document Analysis and Recognition (IJ DAR)*, 10(1):27–38.
- Suderman (2004). The american national corpus first release. pages 81–84.
- Suderman, S. (2010). Anc2go: A web application for customized corpus creation.
- Svartvik, J. (1990). The london corpus of spoken english: Description and research.
- T. Kasar, D. Kumar, M. N. A. P. D. G. and Ramakrishnan, A. G. (2011). Mast: Multi-script annotation toolkit for scenic text. . *ACM*, 14(1):1–8.
- Taylor, H. M. and Karlin, S. (1998). *An Introduction to Stochastic modeling*. Academic Press), Stanford, California.

-
- Thadchanamoorthy, S. (2013). A tamil handwritten database. *In Proc. of ICDAR*, pages 793–797.
- Tomohisa (2014). A database of on-line handwritten mixed objects named: Kondate. *In ICFHR*, pages 369–374.
- U.V. Marti, H. B. (2002). The iam- database: and english sentence database for offline handwriting recognition. *In International Journal on Document Analysis and Recognition*, pages 39–46.
- V., D., P., P., M., A. K., K.P., S., and S., R. (2009). Chunker for tamil. *In 2009 International Conference on Advances in Recent Technologies in Communication and Computing*, pages 436–438.
- Valerio Basile, Johan Bos, K. E. and Venhuizen, N. (2012). Developing a large semantically annotated corpus. *In Proceedings of the International Conference Language Resources and Evaluation Conference (LREC)*, pages 3196–3200, Istanbul, Turkey.
- Velek (2002). The impact of large training data sets on recognition rate of off-line japeese kanji characters clacifiers. *In Workshop DAS*, (106–109).
- Wynne, M. (2005). Stylistics: corpus approaches. pages 1–6.
- Xiao R., R. P. and T., M. (2009). A frequency dictionary of mandarin chinese.
- Yajnik, A. (2017). Part of Speech Tagging Using Statistical Approach for Nepali Text. 11(1):76–79.
- Yin, F., Wang, Q.-F., and Liu, C.-L. (2009). A tool for ground-truthing text lines and characters in off-line handwritten chinese documents. *In Document Analysis and Recognition, 2009. ICDAR '09. 10th International Conference on*, pages 951–955.
- Yuri Lin, Jean-Baptiste Michel, E. L. A. J. O.-W. B. and Petrov, S. (2012). Syntactic annotations for the google books ngram corpus. *In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 169–174. Association for Computational Linguistics.
- Zhou and Su (2002a). *Named Entity Recognition using an HMM-based Chunk Tagger*. Proceedings of 40th Annual Meeting of the Association for Computational Linguistics, Pennsylvania, USA.
- Zhou and Su (2002b). *Named Entity Recognition using an HMM-based Chunk Tagger*. Proceedings of 40th Annual Meeting of the Association for Computational Linguistics, Pennsylvania, USA.

Ziaratban, M., Faez, K., and Bagheri, F. (2009). Fht: An unconstraint farsi handwritten text database. In *Proceedings of the 2009 10th International Conference on Document Analysis and Recognition, ICDAR '09*, pages 281–285, Washington, DC, USA. IEEE Computer Society.