

**MACHINE  
RECOGNITION AND MORPHOLOGICAL  
ANALYSIS OF SUBANTA-PADAS**

*Dissertation submitted to Jawaharlal Nehru University*

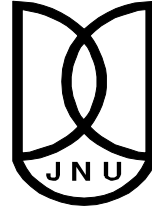
*In partial fulfillment of the requirements*

*For the award of the*

*Degree of*

**MASTER OF PHILOSOPHY**

**SUBASH**



**Special Centre for Sanskrit Studies**

**Jawaharlal Nehru University**

**New Delhi-110067**

**INDIA**

**2006**

**MACHINE  
RECOGNITION AND MORPHOLOGICAL  
ANALYSIS OF SUBANTA-PADAS**

*Dissertation submitted to Jawaharlal Nehru University*

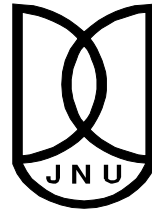
*In partial fulfillment of the requirements*

*For the award of the*

*Degree of*

**MASTER OF PHILOSOPHY**

**SUBASH**



**Special Centre for Sanskrit Studies**

**Jawaharlal Nehru University**

**New Delhi-110067**

**INDIA**

**2006**

***To  
My  
Loving Mummy  
Papa  
Bhaiya  
And  
Bhabhi Ji***

शब्दस्य परिणामोऽयमित्याम्नायविदो विदुः।  
छन्दोभ्य एव प्रथममेतद् विश्वं व्यवर्तत ॥  
एकस्य सर्वबीजस्य यस्य चेयमनेकधा ।  
भोक्तृभोक्तव्यरूपेण भोगरूपेण च स्थितिः॥

## **ACKNOWLEDGEMENT**

---

*I acknowledge with a deep sense of gratitude, continuous help and encouragement from my supervisor Dr. Girish Nath Jha, which he has provided to me despite his busy schedule. Without his able direction, active supervision, positive attitude and motivation, it would have been extremely difficult for the present research and development to complete. He trained me in Java programming and developing database, creating text files and other linguistic resources. He also helped me from time to time in programming, RDBMS techniques, text files and dissertation writing and solved several difficulties.*

*I express deepest thanks and gratitude for Chairperson, Prof. Shashiprabha Kumar and other faculty members - Dr. Hariram Mishra, Dr. Rajnish Kumar Mishra, Dr. Santosh Kumar Shukla and Dr. Ram Nath Jha - of the **Special Center for Sanskrit Studies (SCSS), Jawaharlal Nehru University, New Delhi**, for encouraging and allowing this research to complete.*

*My sincere thanks to my graduation teachers Dr. Vijay Shankar Pandey, Dr. Narendra Nath Yadav and Dr. Vidya Bhushana Kulshreshtha who motivated me for this R & D.*

*I convey my earnest credit to Mummy, Papa, Bhabhi ji, and younger brother Ramesh, whose positive view, love and extra care have made easy to complete this work.*

*I am very grateful to my Bhaiya ji Mr. Vinod Kumar whose positive thoughts, affection, extra care and financial support for me can hardly be repaid.*

*I am also thankful to Radhey Shyam Kumar, Rampujan and Ghanshyam. They have always been good friends and have inspired me for this work.*

*I extend my gratitude to the **University Grants Commission (UGC)** and **Microsoft India Ltd. (MSI)**, who sponsored the projects (in which I worked)- **Online Multilingual Amarakoṣa** and **Devanagari handwriting recognition for tablet PCs** respectively, for providing the financial support.*

*My special thanks for my seniors (Priti, R.Chandrashekhar, and Sudhir K Mishra) and juniors (Sachin, Muktanand, Diwakar Mani, Diwakar Mishra and Surjit Kumar Singh) of the SCSS for helping me systematize and structure the ideas for my R & D.*

*I am also grateful to some of the M.A. students of my centre for entering data and collecting corpus for this research.*

*I express my thanks to colleagues (Renu, Anita, Shruti, Surjyakamal, Abhay), friends from my hostel (Prashant, Payal, Sharad, Sushil, Arup) and some other of my friends (Ram Chandra, Dhananjay Mani, Balawant, Karan, Anil, Arun, Satyaprakash, Akhand Pratap, D.D. Tiwari and Omkar) who helped me from time to time during this work.*

*Lastly, I express my sincere thanks to all the office staff of my centre (Santosh ji, Rita ji, Haribabu ji and Lovely ji) and those who directly or indirectly helped me for completing this work.*

**SUBHASH CHANDRA**

# Contents

---

<b>ACKNOWLEDGEMENT</b>	
<b>List of abbreviations used in the dissertation</b>	<b>01-01</b>
<b>Transliteration key used in the dissertation</b>	<b>02-03</b>
<b>Devanagri input mechanism according to Baraha software</b>	<b>04-04</b>
<b>CONTENT</b>	<b>05-09</b>
<b>INTRODUCTION</b>	<b>10-16</b>
<b>CHAPTER I</b>	<b>17-37</b>
<b>Morphological analyzers and the system of Pāṇini</b>	<b>17-37</b>
<b>1.1 Morphological analyzers</b>	<b>17-27</b>
<b>1.1.1 Approaches and technology</b>	<b>17-18</b>
1.1.1.1 Cut-and-Paste method	17-18
1.1.1.2 Finite State techniques	18-18
<b>1.1.2 Morphological Analyzers for non Indian languages</b>	<b>18-24</b>
1.1.2.1 PC Kimmo	18-20
1.1.2.2 CLAWS	20-20
1.1.2.3 Arabic morphological analysis and generation	20-21
1.1.2.4 Comprehensive morphological analysis of Chinese, Japanese & Korean text	21-21
1.1.2.5 ARIES Natural Language tool for Spanish	21-23
1.1.2.6 Morphological analysis of Bulgarian sentence	23-23
1.1.2.7 French Morphological analyzer	23-23
1.1.2.8 Greek Morphological Analyzer	23-23
1.1.2.9 Latin parser and translator	23-24
1.1.2.10 Multilingual Verb Conjugator	24-24
1.1.2.11 Turkish Morphological analyzer	24-24
<b>1.1.3 Morphological Analyzers for Indian languages</b>	<b>24-26</b>
1.1.3.1 Morphological analyzers by Akshara Bharati Group	25-25
1.1.3.2 Hindi-Marathi-Telugu Morphological analyzers	26-26
<b>1.1.4 Current status</b>	<b>26-26</b>
<b>1.1.5 Sanskrit morphology</b>	<b>26-27</b>
<b>1.2 System of Pāṇini and <i>Subanta</i></b>	<b>27-37</b>
<b>1.2.1 System of Pāṇini</b>	<b>27-35</b>
1.2.1.1 Technical terms of Pāṇini	34-35
<b>1.2.2 Nominal Inflectional Morphology (<i>subanta</i>) of Pāṇini</b>	<b>35-37</b>
1.2.2.1 <i>avyaya subanta</i> ( indeclinable NPs)	36-36
1.2.2.2 Nominal <i>Subanta</i> (base NPs)	36-36

1.2.2.3 <i>samāsānta subanta</i> (compound NPs)	37-37
1.2.2.4 <i>kṛdanta subanta</i> (primary derived NPs)	37-37
1.2.2.5 <i>taddhitānta subanta</i> (secondary derived NPs)	37-37
1.2.2.6 <i>strīpratyayānta subanta</i> (feminine derived NPs)	37-37
<b>CHAPTER II</b>	<b>38-59</b>
<b><i>Subanta</i> types &amp; processes: their recognition</b>	<b>38-59</b>
<b>2.1 Constituents of Sanskrit sentence</b>	<b>38-41</b>
<b>2.1.1 <i>subanta</i> and its morphological types</b>	<b>38-40</b>
2.1.1.1 Primary <i>subanta</i>	38-39
2.1.1.2 Secondary <i>subanta</i>	39-40
2.1.1.2.1 <i>taddhitānta subanta</i>	39-39
2.1.1.2.2 <i>samāsānta subanta</i>	39-40
2.1.1.2.3 <i>strīpratyayānta subanta</i>	40-40
2.1.1.2.4 <i>avyaya subanta</i>	40-40
<b>2.1.2 <i>tiñanta</i> and its morphological types</b>	<b>40-41</b>
2.1.2.1 Primary <i>tiñanta</i>	41-41
2.1.2.2 Derived <i>tiñanta</i>	41-41
<b>2.2 <i>subanta</i> formation processes</b>	<b>41-57</b>
2.2.1 Vowel ending <i>prātipadika</i>	42-52
2.2.2 Consonant ending <i>prātipadika</i>	52-57
<b>2.3 <i>subanta</i> recognition mechanism</b>	<b>57-59</b>
2.3.1 Recognition of punctuations and non- <i>subanta</i> words	57-57
2.3.2 <i>Avyaya</i> recognition	58-58
2.3.3 Verb ( <i>tiñanta</i> ) recognition	58-59
2.3.4 <i>Subanta</i> recognition	59-59
<b>CHATER III</b>	<b>60-89</b>
<b><i>SUBANTA</i> ANALYSIS</b>	<b>60-89</b>
<b>3.1 Vowel ending <i>prātipadika</i></b>	<b>60-72</b>
<b>3.1.1 Nominative singular/dual/plural</b>	<b>60-61</b>
3.1.1.1 Special cases/exceptions	61-61
<b>3.1.2 Accusative singular/dual/plural</b>	<b>62-63</b>
3.1.2.1 Special cases/exceptions	63-63
<b>3.1.3 Instrumental singular/dual/plural</b>	<b>63-64</b>
<b>3.1.4 Dative singular/dual/plural</b>	<b>65-66</b>
3.1.4.1 Special cases/exceptions	66-66
<b>3.1.5 Ablative singular/dual/plural</b>	<b>66-67</b>
3.1.5.1 Special cases/exceptions	67-67
<b>3.1.6 Genitive singular/dual/plural</b>	<b>67-69</b>
<b>3.1.7 Locative singular/dual/plural</b>	<b>69-70</b>
3.1.7.1 Special cases/exceptions	70-70
<b>3.1.8 Vocative singular/dual/plural</b>	<b>71-72</b>
<b>3.2 Consonant ending <i>prātipadika</i></b>	<b>72-86</b>
<b>3.2.1 Nominative singular/dual/plural</b>	<b>72-74</b>
<b>3.2.2 Accusative singular/dual/plural</b>	<b>74-76</b>



3.2.3 Instrumental singular/dual/plural	76-78
3.2.4 Dative singular/dual/plural	78-79
3.2.5 Ablative singular/dual/plural	79-81
3.2.6 Genitive singular/dual/plural	81-83
3.2.7 Locative singular/dual/plural	83-85
3.2.8 Vocative singular/dual/plural	85-86
3.3 Complicated <i>subantas</i> and ambiguity handling	87-89
3.3.1 Ambiguous <i>prātipadika</i>	88-89
3.3.1.1 Ambiguity due to end-character	88-88
3.3.1.2 Morphological ambiguity	88-89
3.3.2 Ambiguous <i>vibhaktis</i>	89-89
3.3.2.1 Always ambiguous <i>vibhaktis</i>	89-89
<b>CHAPTER IV</b>	<b>90-104</b>
<b>ONLINE SUBANTA RECOGNIZER AND ANALYZER</b>	<b>90-104</b>
<b>4.1 Description of SRAS</b>	<b>90-101</b>
4.1.1 Architecture of the system	90-95
4.1.1.1 Multi-layered architecture	90-90
4.1.1.2 The front-end: online interface	91-91
4.1.1.3 The back-end: database / txt files	91-94
4.1.1.4 Database connectivity	94-94
4.1.1.5 The web server	94-95
4.1.1.5.1 Apache Tomcat:	94-94
4.1.1.5.2 Java Servlet Technology	95-95
4.1.1.5.3 Java Server Pages	95-95
4.1.2 Module Description	95-101
4.1.2.1 RSubanta	99-100
4.1.2.1.1 Preprocessor / Recognizer	97-97
4.1.2.1.2 Punctuation checker	97-99
4.1.2.1.3 non-subanta checker	99-100
4.1.2.1.4 Subanta Recognizer	100-100
4.1.2.2 Subanta Analyzer	100-101
4.2. Test Corpora	102-102
4.3 How it works	102-104
4.3.1 Result analysis	104-104
<b>CONCLUSION</b>	<b>105-114</b>
Limitations of the system	105-107
Limitations of the recognition process	105-106
Limitations of the analysis process	106-107
Ambiguity resolution strategies	107-107
Accuracy of results	107-109
Sample: 1	109-109
Output after recognition and analysis of input text	109-111
Processing speed	112-112
Future R&D	112-114

<b>APPENDICES</b>	<b>115-122</b>
APPENDIX- I	115-115
APPENDIX- II	116-116
APPENDIX- III	117-117
APPENDIX- IV	118-118
APPENDIX- V	119-119
APPENDIX- VI	120-120
APPENDIX- VII	121-121
APPENDIX- VIII	122-122
 <b>BIBLIOGRAPHY</b>	 <b>123-135</b>
 <b>FIGURES</b>	
Figure- 1.1	19-19
Figure- 4.1	91-91
Figure- 4.2	92-92
Figure- 4.3	98-98
Figure- 4.4	102-102
 <b>TABLES</b>	
Table-1	12-12
Table-1.1	30-30
Table-1.2	33-34
Table-2.1	58-58
Table-2.2	59-59
Table-3.1	60-61
Table-3.2	61-61
Table-3.3	62-63
Table-3.4	63-63
Table-3.5	64-64
Table-3.6	65-66
Table-3.7	66-66
Table-3.8	66-67
Table-3.9	67-67
Table-3.10	68-69
Table-3.11	69-70
Table-3.12	70-70
Table-3.13	71-72
Table-3.14	73-74
Table-3.15	75-76
Table-3.16	77-78
Table-3.17	78-79
Table-3.18	80-81
Table-3.19	81-83

Table-3.20	83-85
Table-3.21	85-86
Table-3.22	87-88
Table-4.1	88-89
Table-5.1	108-108
<b>(SRAS) CD</b>	139-139

### List of abbreviations used in the dissertation

R&D	=	Research and	3-2	=	तृतीया द्विवचन
		Development	3-3	=	तृतीया बहुवचन
SRAS	=	Subanta	4-1	=	चतुर्थी एकवचन
recognizer		and Analyzer	4-2	=	चतुर्थी द्विवचन
AD	=	Aṣṭādhyāyī	4-3	=	चतुर्थी बहुवचन
SK	=	Siddhāntakaumudī	5-1	=	पञ्चमी एकवचन
KV	=	Kāśikāvṛtti	5-2	=	पञ्चमी द्विवचन
NLP	=	Natural	5-3	=	पञ्चमी बहुवचन
Languages		Processing	6-1	=	षष्ठी एकवचन
AI	=	Artificial	6-2	=	षष्ठी द्विवचन
Intelligence			6-3	=	षष्ठी बहुवचन
NPs	=	subanta-padas	7-1	=	सप्तमी एकवचन
VPs	=	tinanta-padas	7-2	=	सप्तमी द्विवचन
PL	=	पुंलिङ्ग	7-3	=	सप्तमी बहुवचन
NL	=	नपुंसकलिङ्ग	8-1	=	सम्बोधन एकवचन
SL	=	स्त्रीलिङ्ग	8-2	=	सम्बोधन द्विवचन
प्र.	=	प्रथमा	8-3	=	सम्बोधन बहुवचन
द्वि.	=	द्वितीया			
तृ.	=	तृतीया			
च.	=	चतुर्थी			
प.	=	पञ्चमी			
ष.	=	षष्ठी			
स.	=	सप्तमी			
एक.	=	एकवचन			
द्वि.	=	द्विवचन			
ब.	=	बहुवचन			
बहु.	=	बहुवचन			
1-1	=	प्रथमा एकवचन			
1-2	=	प्रथमा द्विवचन			
1-3	=	प्रथमा बहुवचन			
2-1	=	द्वितीया एकवचन			
2-2	=	द्वितीया द्विवचन			
2-3	=	द्वितीया बहुवचन			
3-1	=	तृतीया एकवचन			

### Transliteration key used in the dissertation

अ	=	a
आ	=	ā
इ	=	i
ई	=	ī
उ	=	u
ऊ	=	ū
ऋ	=	ṛ
ॠ	=	ṝ
लृ	=	lṛ
ॠ	=	lṝ
ए	=	e
ऐ	=	ai
ओ	=	o
औ	=	au
अं	=	am
अः	=	ḥ
क	=	k
ख	=	kh
ग	=	g
घ	=	gh
ङ	=	ṅ
च	=	c
छ	=	ch
ज	=	j
झ	=	jḥ
ञ	=	ñ
ट	=	ṭ
ठ	=	ṭh
ड	=	ḍ
ढ	=	ḍh
ण	=	ṇ
त	=	t
थ	=	th
द	=	d
ध	=	dh
न	=	n
प	=	p

फ़	=	ph
ब	=	b
भ	=	bh
म	=	m
य	=	y
र	=	r
ल	=	l
व	=	v
ष	=	ś
श	=	ṣ
स	=	s
ह	=	h
क्ष	=	kṣ
त्र	=	tr
ज	=	j
८	=	‘

Devanagari Input Mechanism according to Baraha software (<http://www.baraha.com>) for SRAS website

VOWELS				
a [अ],	aa/A [आ],	i [इ],	ee [ई],	u [उ],
oo [ऊ],	Ru [ऋ],	RU [ॠ],	IRu [ऌ],	IRU [ॡ],
e [ए],	ai [ऐ],	o [ओ],	au [औ],	aM [अं],
aH [अः]				
CONSONANTS				
k [क],	kh/K [ख],	g [ग],	gh [घ],	~G [ङ],
c [च],	C [छ],	j [ज],	jh/J [झ],	~J [ञ],
T [ट],	Th [ठ],	D [ड],	Dh [ढ],	N [ण],
t [त],	th [थ],	d [द],	dh [ध],	n [न],
p [प],	ph [फ],	b [ब],	bh [भ],	m [म],
y [य],	r [र],	l [ल],	v/w [व],	sh/S [श],
Sh;[ष]	s [स],	h [ह],	kSh [क्ष],	tra [त्र],
j~J [ज्ञ],				

## **INTRODUCTION**



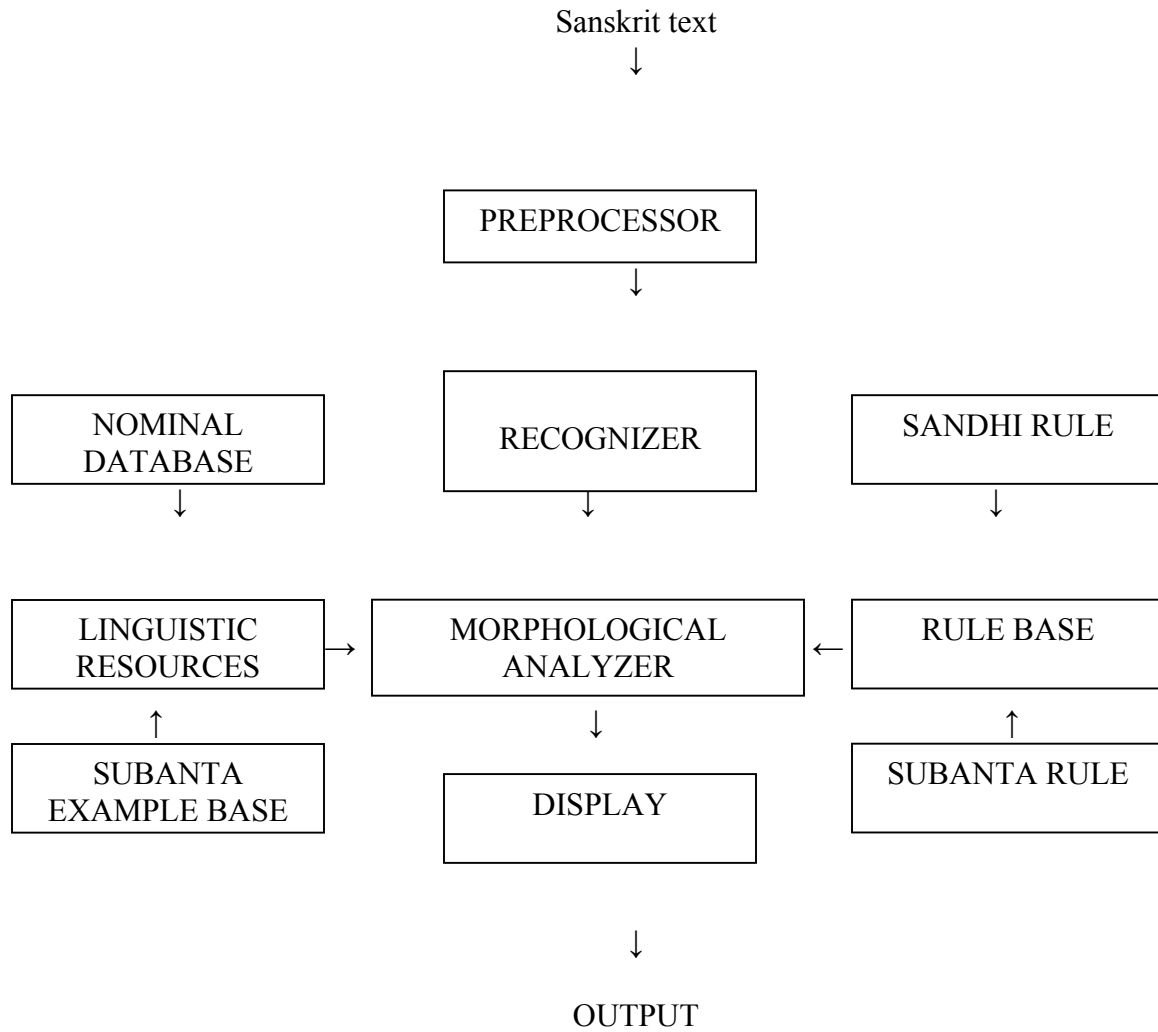
## **Introduction**

The dissertation is an R&D effort at the M. Phil. level for developing a Subanta Recognition and Analysis System (SRAS) based on the *subanta* formulations of Pāṇini. The work consists of the following –

- comprehensive research on the *subanta* rules of Aṣṭādhyāyī (AD), Siddhānta Kaumudī (SK), and Kāśikā Vṛtti (KV),
- formalizing the *subanta* rules of Pāṇini and developing the rule base,
- creating linguistic resources appropriate for *subanta* recognition and analysis
  - creating an example base of pronouns, exceptions, and other typical *subantas*
  - creating a database of basic verb forms of about 90,000 verbs (of 500 commonly occurring verb roots)
  - creating an *avyaya* list of 520 entries
  - creating a test corpus of 120 electronic files of modern Sanskrit usage
  - creating a testing corpus of 500 common verb roots with added *lyap* primary suffix ending
  - creating a testing corpus of 500 common verb roots with added *ktvā* primary suffix ending
  - creating a testing corpus of 500 common verb roots with added *tumun* primary suffix ending
  - creating a corpus of Pañcatantra stories
  - creating another corpus of some stories, essay and other from Sanskrit magazines and other resources
- studying the feasibility of a Unicode online system for *subanta* recognition and analysis

- evaluating the tools and techniques used – JSP for front end, Java for servlet objects, and MS-SQL Server 2005 for backend, JDBC for connectivity and Apache-Tomcat for web server
- creating two systems – one for running in a server based non-portable environment with RDBMS as back-end, and other for stand-alone like environment but on Tomcat/Apache web server

The SR&AS work-flow can be understood in the following diagram –



The SR&AS gives 90 % of result when tested on the corpus. The result analysis was done on the following corpus (UTF-8, Devanagri) –

S.No.	File	Theme	Source	Words	Time in sec.
1	Corpus-1	rājā sagaraḥ	andeśaḥ	609	3
2	Corpus-2	samrāṭa aśokaḥ	sandeśaḥ	916	3.2
3	Corpus-3	ekaḥ nibandhaḥ	sandeśaḥ	882	3
4	Corpus-4	cācā neharuḥ	sandeśaḥ	332	1
5	Corpus-5	sarasvatī vandanā and a story	sandeśaḥ	241	1
6	Corpus-6	ādhunika praśāsanaḥ	sandeśaḥ	1045	3.5
7	Corpus-7	ekaḥ vaṇikaḥ	sandeśaḥ	849	2
8	Corpus-8	paśya me rūpāṇi	sandeśaḥ	1328	4
9	Corpus-9	Sanskrit sikṣā	sandeśaḥ	306	2
10	Corpus-10	saṅghe śaktiḥ	sandeśaḥ	4207	6

Table- 1 : test corpus

## Previous work

The Indian Heritage Group of the Centre for Development of Advanced Computing (CDAC) has developed a system called *DESIKA*, which claims to process all the words of Sanskrit and includes generation and analysis (parsing). It claims to have an exhaustive database based on *amarakoṣa*, a rule-base using the grammar rules of Pāṇini's aṣṭādhyāyī and heuristics based on *nyāya & mimāṃsā śāstras* for semantic and contextual processing. However, the system (as available at the TDIL site<sup>1</sup>) has *subanta* generation only and even that does not work properly.

Huet has developed a Grammatical Analyzer System, which tags *subanta-padas* by analyzing *sandhi*, *samāsa* and *sup* affixation. This system is available online at:

<sup>1</sup> <http://tdil.mit.gov.in/download/Shabdabodha.htm>

<http://pauillac.inria.fr/~huet/SKT/sanskrit.html>. The system suffers from weaknesses in terms of not being rooted in the Pāṇini's system. As a result, there are so many errors that it practically becomes unusable. Secondly, the Huet's system takes phrases and not full sentences or texts.

The Rashtriya Sanskrit Vidyapeeth, Tirupathi under the leadership of Prof. K. V. Ramakrishnamacharyulu (currently Vice Chancellor of Rajasthan Sanskrit University) has done commendable work on the Sansk-net project. This Project was proposed by the Indian Heritage Group (IHG), Real-Time Systems Group (RTSG), and Center for Development of Advanced Computing (C-DAC), Bangalore, to be an initiative with Rashtriya Sanskrit Vidyapeetha (RSVP), Tirupati. The objectives of this project are- to present the database available in different institutions in a computer framework, develop the hardware, software and the technical capability to place the information in the modern technical framework, computer linkage among different institutions so that each institution can have access to the database available in the other institutions, make use of the principles and techniques available in *nyāya*, *vyākaraṇa*, *vedānta* and *vedāṅga* for developing new paradigms for the computer, packages for training for the faculties in the scientific work and *śāstraic* world for making best use of the infrastructural facility and facilitate preservation of the information on rare manuscript, Vedic literature and *śāstras*. Prof. Vineet Chaitanya and Amba Kulkarni are visiting the institution and are currently guiding several Sanskrit R&D initiatives with far reaching consequences.

Vanasthali Vidyapeeth, Vanasthali, Rajasthan, has also been working on Sanskrit. Jawaharlal Nehru University finished the CASTLE (Computer Assisted Sanskrit Teaching and Learning Environment) project and some related work in this area like Sanskrit processing tools and Sanskrit authoring system. Some of these may be available on the TDIL website <http://tdil.mit.gov.in>.

Academy of Sanskrit Research, Melkote, Mysore has been actively involved in bringing scholars doing technology R&D for Sanskrit and *śāstras* on a single platform. In 1993, it organized a seminar on *Sanskrit and computer based linguistics* and in 1994, a seminar on *Interface Mechanisms in śāstras and Computer Science*. The latter, among other things, brought out similarities in the traditional Indian theories and principles of Artificial Intelligence.

The Special Centre for Sanskrit Studies, Jawaharlal Nehru University, New Delhi is currently engaged in the following R&D - *kāraka* Analyzer, *sandhi* splitter and analyzer, verb analyzer, NP gender agreement, POS tagging of Sanskrit, online Multilingual *amarakoṣa*, Pāṇini's Aṣṭādhyāyī search engine, online *Mahābhārata* indexing and Jha (2006) presented a model of Sanskrit Analysis System (SAS)<sup>2</sup>. The RCILTS project under Prof. G.V. Singh at the School of Computer and Systems Sciences has prepared useful linguistic resources for Sanskrit.

Morphological analyzers for Sanskrit, Telugu, Hindi, Marathi, Kannada and Punjabi have been developed by Akshara Bharathi Group at Indian Institute of Technology<sup>3</sup>, Kanpur, and University of Hyderabad funded by Ministry of Information Technology the project claims to have 95% coverage for Telugu (arbitrary text in modern standard Telugu), and 88% coverage for Hindi. This system is available on the site for downloading as well as online at: <http://www.iiit.net/ltrc/morph/index.htm>

Anusaaraka (developed by Akshar Bharati group, IIIT, Hyderabad) is a computer software which renders text from one Indian language into another, a sort of machine translation. It produces output which is comprehensible to the reader, although at times it might not be grammatical. The system is available at the IIIT Hyderabad site<sup>4</sup>)

## How is this work different?

The work is different from existing research in the following ways –

1. no online RDBMS based recognizer-analyzer is available till date, which accepts and displays results in Unicode Devanagari script but this system takes Unicode Devanagari text and displays results in Devanagari,
2. this system takes Devanagari utf-8 text as input and delivers Devanagari utf-8 text output using a Java servlet – Apache-Tomcat - JDBC - RDBMS technology,

---

<sup>2</sup> Jha, Girish Nath et al, “Towards a Computational analysis system for Sanskrit” in the proceeding of first National symposium on Modeling and Shallow parsing of Indian Languages at Indian Institute of Technology Bombay pp 25-34 on 2<sup>nd</sup> to 4<sup>th</sup> April 2006

<sup>3</sup> <http://www.iiit.net/ltrc/morph/index.htm> access on Feb 13th 2006

<sup>4</sup> [http://www.iiit.net/ltrc/Anusaaraka/anu\\_home.html](http://www.iiit.net/ltrc/Anusaaraka/anu_home.html)

3. gives a comprehensive computational analysis of *subanta-padas* in a Sanskrit text, and does basic tagging of verbs and *avyayas* too,
4. uses a hybrid approach of Pāṇinian formalism and example-based techniques to process input text. It works on the morphological nature of bases and applies the *vibhakti* information for processing,
5. the system can be used for larger processing of Sanskrit for text simplification and machine translation

## Summary of chapters

**Chapter I** discusses morphological analyzers, current status of R&D in this field, structure and organization of of Aṣṭādhyāyī (AD), and *subanta* of Pāṇini.

**Chapter II** discusses *subanta* formalism of Pāṇini and mechanisms to recognize verb, *avyaya* and *subanta* in Sanskrit text.

**Chapter III** discusses the analysis of *subanta-padas*.

**Chapter IV** discusses the implementation aspects – the front end, Java objects, databases, linguistic resources (corpus and rule bases and example bases), how they work and what is basic requirement of the system and how to apply *sandhi* and *subanta* rule where ever necessary.

**Conclusion** discusses future R&D, limitations of the system and result analysis.

## Limitations

- Some verbs have the same form as *subantas*, for example भवति, रामः, गच्छति the system will exclude such *subantas* as verbs.
- The morphological ambiguity of several *vibhaktis* like भ्याम्, ओस्, भ्यस् persist when processed in isolation. These can be solved at the larger level.
- It does not split the *samāsas* into constituent *subantas* by way of reverse *sandhi* so that reverse *subanta* can be done. This will be implemented with the *samāsas* component.
- In some cases, the recognition of the base form is ambiguous, for example, for the ह् ending and श् ending *prātipadikas*, the last characters change in ट् and ड् respectively. So

system ca not recognize correct *prātipadikas*. In this condition, the system will give other possible results. For example: लिट्, लिङ्, तादृक्, तादृग् etc.

## **Chapter-I**

# **MORPHOLOGICAL ANALYZERS AND THE SYSTEM OF PĀṆINI**



## 1.1. Morphological analyzers

Morphological analyzers are critical for any useful natural language system. This is more so for morphology rich Indian languages. Even for configurational languages like English, a syntactic parse will have to go down further to the morphology level. The variation and arbitrariness in the behavior of morphemes have been challenges for most analyzers. Traditionally morphological processors have been used for the following tasks—

- Analysis
  - Taggers
  - Chunkers
  - Word breakers/lemmatizers
- Word builders
  - Paradigm generator

Recent advancements in this field for building useful analyzers have been in last 10-15 years. The field has become more applied now than being just academic research.

### 1.1.1 Approaches and technology

Computational morphology has been following simple rule based string processing techniques as well as finite state techniques depending on the nature of morphology in each language.

#### 1.1.1.1 Cut and Paste method

Cut and paste is a very popular method in computational linguistics. The canonical form is derived by removing and adding letters to the end of a string. The best known ancestor of these systems dates back to the 1960s<sup>5</sup>.

Another system known as **MORPHOGEN** (Petheroudakis, 1991) is a commercial toolkit for creating sophisticated cut and paste analyzers<sup>6</sup>. The **MAGIC** (Schuller, Zierl, 1993) is a cut

---

<sup>5</sup> Allen, J., Hunnicutt, M. S., and Klatt, D. (1987). *From text to speech---the MITalk system*. MIT Press, Cambridge, Massachusetts.

<sup>6</sup> Petheroudakis, J. (1991). *MORPHOGEN automatic generator of morphological information for base form reduction*. Technical report, Executive Communication Systems ECS, Provo, Utah.

and paste rule based system in which rules are applied in advance to produce the right allomorph for every allowed combination of a morpheme<sup>7</sup>

### 1.1.1.2 Finite State techniques

Finite state techniques are used in cases where large lexicons are to be checked. It also explains morphotactics better than the cut-paste method. Automatic recognition and generation of word forms was introduced early 80s. Rules of morphological alternations could be implemented using FSTs as a finite state network (Johnson 1972, Kaplan and Kay 1994)<sup>8</sup>. First practical application of model appeared in the 90s (Koskenniemi 83, Karttunen 1993, Antworth 1990, Karttunen and Beesley 1992, Ritchie, Russell et al, 1992, sproat 1992)<sup>9</sup>. These systems used linked letter trees for the lexicon and parallel FSTs encoding morphemic alternations. The FS techniques are generally used for searching large scale spellchecking wordlists. They also allow bi-directional processing (i.e. both generation and analysis can be performed)

### 1.1.2 Morphological Analyzers for non Indian languages

Some morph-analyzers have been developed of foreign languages also. Many analyzers available on web, some of these details follow:

#### 1.1.2.1 PC Kimmo

PC-KIMMO is a new implementation for microcomputers of a program dubbed KIMMO after its inventor Kimmo Koskenniemi (Koskenniemi 1983). It is of interest to computational linguists, descriptive linguists, and those developing natural language processing systems<sup>10</sup>. The program is designed to generate (produce) and/or recognize (parse) words using a two-level model of word structure in which a word is represented as a correspondence between its lexical level form and its surface level form. Work on PC-KIMMO

---

<sup>7</sup> Schuller, G., Zierl, M., and Hausser, R. (1993). *MAGIC. A tutorial in computational morphology*. Technical report, Friedrich-Alexander Universitat, Erlangen, Germany.

<sup>8</sup> Kaplan, Ronald M., and Kay, Martin (1981). *Phonological Rules and Finite-State Transducers*. Paper presented at the Annual Meeting of the Linguistic Society of America. New York.

<sup>9</sup> Kimmo Koskenniemi. *Two-level morphology: A general computational model for word form recognition and production*. Publication No: 11, Department of General Linguistics, University of Helsinki, 1983.

<sup>10</sup> <http://www.sil.org/pckimmo/> accessed on 25th December 2005

began in 1985. A PC-KIMMO description of a language consists of two files provided by the user:

- a **rules file**, which specifies the alphabet and the phonological (or spelling) rules, and
- a **lexicon file**, which lists lexical items (words and morphemes) and their glosses, and encodes morph tactic constraints.

The theoretical model of phonology embodied in PC-KIMMO is called two-level phonology. The two functional components of PC-KIMMO are the generator and the recognizer. The generator accepts as input a lexical form, applies the phonological rules, and returns the corresponding surface form. It does not use the lexicon. The recognizer accepts as input a surface form, applies the phonological rules, consults the lexicon, and returns the corresponding lexical form with its gloss.

Figure 1 shows the main components of the PC-KIMMO system<sup>11</sup>.

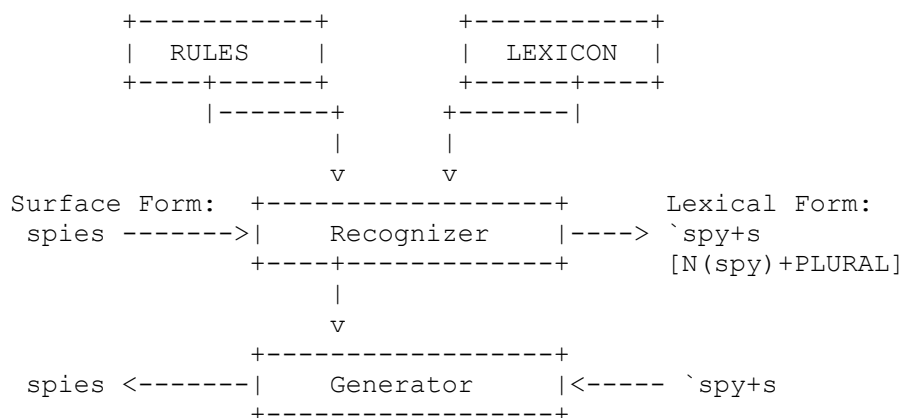


Figure 1.1 Main components of PC-KIMMO

PC-KIMMO runs on the Windows, Macintosh and UNIX systems. There are two versions of the PC-KIMMO release, one for IBM PC compatibles and one for the Macintosh. Each contains the executable PC-KIMMO program, examples of language descriptions, and the source code library for the primitive PC-KIMMO functions. The PC-KIMMO executable program and the source code library are copyrighted but are made freely available to the general public under the condition that they not be resold or used for commercial purposes.

<sup>11</sup> <http://www.sil.org/pckimmo/>

The PC-KIMMO release contains the executable PC-KIMMO program, the function library, and examples of PC-KIMMO descriptions for various languages, including English, Finnish, Japanese, Hebrew, Kasem, Tagalog, and Turkish. These are not comprehensive linguistic descriptions; rather they cover only a selected set of data<sup>12</sup>.

### 1.1.2.2 CLAWS

CLAWS (Constituent Likelihood Automatic Word-tagging System)<sup>13</sup> POS tagging software for English text has been continuously developed by University Centre for Computer Corpus Research on Language (UCREL) in early 1980s. The latest version of the tagger, CLAWS-4, was used to POS tag 100 million words of the British National Corpus (BNC). CLAWS, has consistently achieved 96-97% accuracy (the precise degree of accuracy varying according to the type of text). Judged in terms of major categories, the system has an error-rate of only 1.5%, with c.3.3% ambiguities unresolved, within the BNC. More detailed analysis of the error rates for the C5 tagset in the BNC can be found within the BNC Manual<sup>14</sup>.

### 1.1.2.3 Arabic Morphological Analysis and Generation

Kenneth R. Beesley at the Xerox Research centre Europe, chemin de Maupertuis, 38240 MEYLAN, France has been developed an Arabic Morphological Analysis and Generation. Arabic morphological analyzer and generator, which was built using [Xerox Finite-State Technology](#). The system accepts Modern Standard Arabic words and returns morphological analyses and English glosses. Arabic words are displayed in Arabic script using Java applets. This lexicography and Arabic-language consultation for the research system was provided by Tim Buckwalter. The Arabic ["Yarb" font](#) used in the interfaces was created by [Yannis Haralambous](#). System design, interfaces, Arabic-script rendering, and computational linguistics were done by [Ken Beesley](#). Arabic language consulting and lexicography for the commercialization in 2002 was provided by Martine Pétrod<sup>15</sup>

---

<sup>12</sup> <http://www.sil.org/pckimmo/>

<sup>13</sup> [http:// www.comp.lancs.ac.uk/ucrel/claws](http://www.comp.lancs.ac.uk/ucrel/claws) access on 15<sup>th</sup> January 2006

<sup>14</sup> [www.comp.lancs.ac.uk/ucrel/claws](http://www.comp.lancs.ac.uk/ucrel/claws)

<sup>15</sup> <http://www.xrce.xerox.com/competencies/content-analysis/arabic/>

#### 1.1.2.4 Comprehensive morphological analysis of Chinese, Japanese and Korean text

Asian language analyzers are used in some of the world's most transaction-heavy environments, like Google's search engine and Amazon's e-commerce site. Rosette Base Linguistics for Chinese, Japanese and Korean are extremely accurate and reliable solutions to help complex applications process unstructured Asian language text by conquering some of these languages' many challenges, such as the use of numerous scripts and absence of spaces between words. Using advanced morphological analysis, Asian Base Linguistics perform functions critical for analyzing Asian text such as segmentation, lemmatization, noun decomposing, part-of-speech tagging, sentence boundary detection, and base noun phrase analysis<sup>16</sup>.

#### 1.1.2.5 ARIES Natural Language Tools for Spanish

The ARIES Natural Language Tools make up a lexical platform for the Spanish language. These tools can be integrated into NLP applications. They include: a large Spanish lexicon, lexical maintenance, access tools, and morphological analyzer and generator. Non-exclusive, non-transferable licenses are available for the following components:

- **The Prolog GRAMPAL analyzer/generator**

A public domain demonstration system written in Prolog of morphological treatment and lexicon. It includes a small demo lexicon, a DCG grammar for word formation and some predicates to test both analysis and generation. It runs under Sicstus Prolog 2.1.9.

- **The Prolog GRAMPAL dictionary**

A collection of Prolog predicates suitable for use with the public domain GRAMPAL DCG grammar. It is capable of generating/recognizing well formed inflected forms for verbs, nouns and adjectives. It has no adverbs, determiners, conjunction, prepositions, etc. It does not treat clitic pronoun attachment or derivatives.

- **The expanded ARIES dictionary**

---

<sup>16</sup> <http://www.basistech.com/base-linguistics/asian/>

A collection of expanded entries (allomorphs) with morphological information. It contains a full set of morphemes dealing with clitic pronoun attachment (but without verb marking for correct attachments). It includes information about some derivative morphological processes (inflected adjectives from past participles and adverbs ended in "-mente" from adjectives).

- **The source ARIES lexical base**

A collection of inflectional models, rules for off-line computing of allomorphs, unexpanded lemma entries, lexicalized irregular words. It is the most complete source of information we have available and the most useful for dictionary maintenance. A tool for expanding the source dictionary to the expanded dictionary is also provided. The current size of this lexicon is 38,500 lemma entries (21,000 nouns, 10,000 adjectives, 7,500 verbs and 500 auxiliary words) plus more than 600 inflectional morphemes.

- **Access tools**

The C/C++ programming interface for lexical access to the ARIES dictionary: It is a set of tools and libraries to build tree indexes to the allomorph dictionary and to retrieve them by an application.

- **Morphological analyzer**

The C/C++ morphological analyzer that makes use of the lexical interface mentioned above. This permits to improve efficiency by integrating word

segmentation with lexical access also. By now, it is a (pseudo)-unification chart based parser for context-free morphological grammars.

These system supported platforms are UNIX and DOS Operating System with GNU gcc/g++ (djgpp for DOS) compilers. Tools have been tested MSDOS, HP-UX 9.05, SunOs 4.1.3 and Solaris 2.4<sup>17</sup>.

### **1.1.2.6 Morphological analysis of Bulgarian sentence**

---

<sup>17</sup> [www.mat.upm.es/~aries/description.html](http://www.mat.upm.es/~aries/description.html)

For a couple of years a system of machine dictionaries a lexical database for the Bulgarian language and a morphological processor have been developed at the Department of Computer Science of the University of Plovdiv. The system for lexical database management is situated on a WEB server<sup>18</sup>.

#### **1.1.2.7 French Morphological analyzer**

The ARTFL Project: morphological analysis using the INFL analyzer allows you to enter one or more French words (lower case only, no punctuation) at the prompt and returns the context-free morphological analysis for each<sup>19</sup>.

#### **1.1.2.8 Greek Morphological Analyzer**

The Perseus Project: morphological analysis using the morphological analyzer allows you to enter one or more Greek words in Latin transliteration at the prompt and returns the morphological analysis for each term<sup>20</sup>.

#### **1.1.2.9 Latin parser and translator 0.96**

This is a Visual Basic program which translating from Latin into English developed by Adam McLean. The alternative translations for ambiguous words have been extended, and the user can now edit, within the program, the Latin as well as English translation files. Version 0.96 has fixed a number of minor bugs in the Latin parsing, added some extra help information, changed the program layout a little, and now comes with an installer<sup>21</sup>.

#### **1.1.2.9 Multilingual Verb Conjugator**

Multilingual verb conjugator, conjugate at least some of the regular verbs in 27 different languages. Database of this system have been cover 27 different languages including Italian,

---

<sup>18</sup> <http://www.uni-plovdiv.bg/dcs/morphe.htm>

<sup>19</sup> [http://humanities.uchicago.edu/orgs/ARTFL/forms\\_unrest/analyze.query.html](http://humanities.uchicago.edu/orgs/ARTFL/forms_unrest/analyze.query.html)

<sup>20</sup> <http://www.perseus.tufts.edu/cgi-bin/morphindex>

<sup>21</sup> [http://www.logosconjugator.org/newverb/verba\\_dba.verba\\_main.create\\_page?lang=en](http://www.logosconjugator.org/newverb/verba_dba.verba_main.create_page?lang=en)

French, Spanish, Polish, German, Esperanto, English, Latin, Portuguese, Greek, Finnish, Czech, Croatian, Sicilian. It is a work in progress that still lacks many verbs<sup>22</sup>.

#### **1.1.2.10 Turkish Morphological analyzer**

This analyzer has been developed using the two-level transducer technology developed by Xerox. It can process near about 900 forms per second. This implementation of Turkish uses about 30,000 Turkish root words.

### **1.1.3 Morphological Analyzers for Indian languages**

The R & D for morphological analyzers for Indian languages was spearheaded by setting up of RCILTS (Resource Center for Indian Languages Technology Solutions) by the Ministry of Communications and Information Technology (MCIT)<sup>23</sup>. The leading resource centers are –

- IIIT, Hyderabad
- IIT Kanpur
- IIT Kharagpur
- IIT Mumbai

Much work in the area of NLP in India has been carried out is still on at several places and in several languages. Of particular mention is the works carried out in

- National Centre for Software Technology (NCST)
- Indian Statistical Institutes (ISI)
- Thapar Institute of Engineering and Technology
- Utkal University, Anna University
- Chennai, Bhubaneshwar,
- A tagged text corpus developed from using the web as source of data in Bengali at Jadavpur University,
- Kolkata and University of Hyderabad,
- Indian Institute of Science, (IISc) Bangalore
- Central Electronics Engineering Research Institute (CEERI), Pilani

---

<sup>22</sup> <http://www.nlp.cs.bilkent.edu.tr/cgi-bin/tma>

<sup>23</sup> <http://www.mcit.gov.eg/>



- Tata Institute of Fundamental Research, Mumbai
- An analyzer being developed for Manipuri
- IBM, India research lab, Microsoft India, Tata Consultancy Services, HP, HCL and Webdunia etc.

Work on Sanskrit informatics has been going on at following place

- C-DAC (Pune)
- Special centre for Sanskrit studies, Jawaharlal Nehru University, New Delhi
- Vanashtali Vidyapeeth, Rajasthan
- Rastriya Sanskrit Vidyapeeth Tirupathi
- Lal Bahadur Shastri Rastriya Sanskrit Vidyapeeth, New Delhi.
- Academy of Sanskrit Research, Melkote, Mysore

#### **1.1.3.1 Morphological Analyzers by Akshara Bharathi Group**

Morphological analyzer for Sanskrit, Telugu, Hindi, Marathi, Kannada and Punjabi have been developed by Akshara Bharathi group at Indian Institute of Technology, Kanpur, India and University of Hyderabad, Hyderabad, India (funded by Ministry of Information Technology, India) and claim for the 95% coverage for Telugu (for arbitrary text in modern standard Telugu) and 88% coverage for Hindi.

#### **1.1.3.2 Hindi-Marathi-Telugu Morphological analyzers<sup>24</sup>**

This morphological analyzer allows you to choose your language and your font. It includes links to pages with other linguistic resources for Indian languages and English, dictionaries the National Institute of Information Technology in India.

#### **1.1.4 Current status**

---

<sup>24</sup> [http://www.iiit.net/ltrc/morph/morph\\_analyser.html](http://www.iiit.net/ltrc/morph/morph_analyser.html)

The Resource Centre for Indian Languages Technology solution Indian Institute Technology Guwahati has developed two morphological analyzers for Assamese and Manipuri. They have been used both in the spell checker and OCR systems. Both the morphological analyzers use the technique of stemming where in the affixes are either deleted or added to arrive at the root words. The system is not available site but detail is available on the IIT Guwahati site: [http://www.iitg.ernet.in/rcilts/r\\_d.htm](http://www.iitg.ernet.in/rcilts/r_d.htm)

### 1.1.5 Sanskrit morphology

The study of the structure and form of words in languages or a language, including inflection, derivation, and the formation of compounds is called morphology. In Sanskrit, a syntactic unit is called *pada*. **Cordona<sup>25</sup> (1988)** posits the formula for Sanskrit sentence (N-E<sup>n</sup>)p...(V-E<sup>v</sup>)p. *Pada* can be nominal (*subanta*) or verbal (*tinanta*). These forms are formed by inflecting the stems and hence they are part of Sanskrit inflectional morphology. The derivational morphology in Sanskrit studies primary forms (*kṛdanta*) and secondary forms (*taddhitānta*), compounds (*samāsa*), feminine forms (*strī pratyayānta*) etc.

Sanskrit has two types of morphology- nominal and verbal. Sanskrit has approximately 2014 verb roots (including *kandvādi*), classified in 10 *gaṇas*, the derived verb forms can have 12 derivational suffixes<sup>26</sup>. These can have *ātmanepadi* and *prasmaipadi*. A verb root may have approximately 2190 (tense, aspect, number etc.) morphological forms. Sanskrit Nominal morphology of two types, Primary [*kṛdanta* (roots forms that end with *kṛt* suffixes)] and secondary [*taddhitānta* (noun forms that end with *taddhita* suffixes)]. Secondary nominal morphology may be of following types like- *samāsānta* (compound nouns), *strīpratyayānta* (feminine forms) etc. They can also include *upasargas* (prefix) and *avyayas* (indeclinables) etc. According to Pāṇini, there are 21 morphological suffixes (seven *vibhaktis* and combination of three numbers = 21)<sup>27</sup> which are attached to the nominal bases (*prātipadika*)<sup>28</sup> according to syntactic category, gender and end character of the base.

<sup>25</sup> **George Cardona, 1988** Pāṇini, His Work and its Traditions, vol ... i (Delhi: MLBD, 1988)

<sup>26</sup> सन्-क्यच्-काम्यच्-क्यङ्-क्यषोऽथाऽऽचारक्विब्-णिज्यङस्तथा ।

यगाय ईयङ् णिङ् चेति द्वादशाऽमी सनादयः ॥

<sup>27</sup> स्वौजसमौट्छष्टाभ्याम्भिस्ङेभ्याम्भ्यस्ङसिभ्याम्भ्यस्ङसोसौङ्योस्सुप् ।४।१।२॥

<sup>28</sup> अर्थवधातुप्रत्ययः प्रातिपदिकम् ।१।२।४५॥, कृत्तद्धितसमासाश्च ।१।२।४६॥

## 1.2. System of Pāṇini and *Subanta*

### 1.2.1 System of Pāṇini

Pāṇini's grammar AD (approximately 7th BCE) is important for linguistic computation for two reasons. One, it provides a comprehensive and rule based account of a natural language in about 4000 rules - the only complete grammatical account of any language so far. Two, the model of a 'grammar-in-motion' that it provides seems to closely mimic a fully functional Natural Language Processing (NLP) system<sup>29</sup>

*Aṣṭādhyāyī* (7<sup>th</sup> BCE) is a composite text including the following modules -

1. *śivasūtra* or *pratyāhārasūtra* (14) (PS)
2. *śabdānuśāsana* or *sūtrapāṭha* (3965 or 3983 in *kāśikāvṛtti*) (SP)
3. *dhātupāṭha* (1967 verb roots - 2014 including *kaṇḍvādi* roots) (DP)
4. *gaṇapāṭha* (other pertinent items like primitive nominal bases, *avyayas*) (GP)

*Sūtrapāṭha* (SP) is arranged in eight *adhyāyas* (chapters) each divided into four sub-chapters (*pādas*). The SP has approximately 3965 rules (*sūtras*) which have been arranged in x.x.x format (to be accessed in as *adhyāya . pāda . sūtra* format)<sup>30</sup>

The following is a summary of topics discussed in the *Aṣṭādhyāyī*<sup>31</sup> -

#### Chapter I

- Major definitional and interpretational rules
- Rules dealing with extension (*atideśa*)
- Rules dealing with *atmanepada-parasmaipada*
- Rules dealing with the *kāraka*

#### Chapter II

- Rules dealing with compounds (*samāsa*)

<sup>29</sup> <http://www.languageinindia.com/feb2004/panini.html>

<sup>30</sup> Jha Girish Nath 'The System of Panini' Language in India, volume 4:2 February 2004

<sup>31</sup> Sharma, Rama Nath, The *Aṣṭādhyāyī* of Pāṇini – Volume-I page-75-76

- Rules dealing with nominal inflection
- Rules dealing with number and gender of compounds
- Rules dealing with replacements relative to roots
- Rules dealing with deletion by *luk*

### Chapter III

- Rules dealing with derivational of roots ending in affixes *san* etc.
- Rules dealing with the derivational of ending in a *kṛt*
- Rules dealing with the derivational of ending in a *tin*

### Chapter IV

- Rules dealing with derivation of a *pada* ending in a *sup*
- Rules dealing with feminine affixes
- Rules dealing with the derivational of nominal stems ending in an affix termed *taddhita*

### Chapter V, VI & VII

- Rules dealing with doubling
- Rules dealing with *samprasāraṇa*
- Rules dealing with the *saṁhitā*
- Rules dealing with the augment (*āgama*) *suṭ*
- Rules dealing with accents
- Rules dealing with phonological operations relatives to a pre-suffix base (*aṅga*)
- Rules dealing with operations relative to affixes augment etc.

### Chapter VIII

- Rules dealing with doubling (*dvitva*) relative to a *pada*
- Rules dealing with accent relative to a *pada*
- Rules dealing with other phonological relatives to a *pada*
- Rules dealing with miscellaneous operations relative to a non-*pada*

Kapoor(1992) has reduced the treatment of subject matter into four divisions<sup>32</sup>: Chapters 1-2 dealing with classification and enumeration of bases and categories, Chapters 3-5 consist of *prakṛti-pratyaya* enumeration, and derivation of bases, Chapters 6-8.1 deal with the synthesis of *prakṛti-pratyaya*, and Chapters 8.2-8.4 deal with the rules of morphophonemic.

### The modules of AD

---

<sup>32</sup> Kapoor, Kapil, “Text Interpretation: The Indian Tradition”

### *śiva sūtras or pratyāhāra sūtra (PS)*

The purpose of the PS component *is* to give a list of all Sanskrit [phonemes](#). But rather than listing just the phonemes, the sūtras in the PS component are interspersed by meta-linguistic markers, called *anubandhas*. By a well defined method (*ādirantyena sahetā*), Pāṇini creates variables or macros to be used in his grammar. For example, *al* refers to the list of all phonemes, *ac* refers to all vowels, *hal* to all consonants and *ṇam* to all nasals. The 14 sutras are-

a i u ṇ	[अ इ उ ण्]
ṛ ḷ k	[ऋ लृ क्]
e o ṅ	[ए ओ ङ्]
ai au c	[ऐ औ च्]
h y v r ṭ	[ह य व र ट्]
l ṇ	[ल ण्]
ñ m ṇ ṇ n m	[ञ म ङ् ण न म्]
jh bh ñ	[झ भ ञ्]
gh ḍh dh ṣ	[घ ढ ध ष्]
j b g ḍ d ś	[ज ब ग ङ द श्]
kh ph ch ṭh th c ṭ t v	[ख फ छ ठ थ च ट त व्]
k p y	[क प य्]
ś ṣ s r	[श ष स र्]
h l	[ह ल्]

The first 4 *sūtras* cover all the vowels and the last 10 *sūtras* include all the consonants. Again, all vowels and consonants of Sanskrit have been arranged in such a way in these *sūtras* that they can be referred to without mentioning them separately.

Of the hundreds *pratyāhāras* that could in principle be formed from these *sūtras*, [Pāṇini](#) has used 43 (of a 44<sup>th</sup> introduced by later grammarians, *rañ=(r,l)* )<sup>33</sup>. Note that some *pratyāhāras*

<sup>33</sup> उणट्ज्वात् स्मृतो हृयेकः, चत्वारश्च चमान्मताः ।  
शलाभ्यां षड् यरात्पञ्च, षाद् द्वौ च कणतस्त्रयः ॥  
केषाञ्चिच्च मते रोऽपि, प्रत्याहारोऽपरो मतः ।  
लस्थाऽवर्णेन वाञ्छन्त्यनुनासिकबलादिः ॥

are ambiguous. For example, *ṇ* occurs twice in the list, which means that you can assign two different meanings to *prtyāhāras aṇ*<sup>34</sup> (including or excluding *ṛ* etc.)

### ***sūtrapāṭha* (SP)**

The SP contains about 3965 *sūtras* or 3983 in *kāśikā vṛtti* arranged in chapters (*adhyāya*) and sub-chapters (*pāda*) in a particular order<sup>35</sup>.

Chapter	Pāda I	Pāda II	Pāda III	Pāda IV	Total Rules
1 <sup>st</sup>	74	73	93	109	349
2 <sup>nd</sup>	71	38	73	85	267
3 <sup>rd</sup>	150	188	176	117	631
4 <sup>th</sup>	176	144	166	144	630
5 <sup>th</sup>	135	140	119	160	554
6 <sup>th</sup>	217	198	138	175	728
7 <sup>th</sup>	103	118	119	97	437
8 <sup>th</sup>	74	108	119	68	369
Total Rules in Aṣṭādhyāyī			3965		

**Table 1.1 : Distribution of AD *sūtras***

*sūtras* are verb-less sentences unlike those in natural language and give an impression of formula or program like code. They are of following types<sup>36</sup> –

- ***Samjña* (Technical Rules):** Rules which assign a particular term to a given entity.
- ***Paribhāṣā* (Interpretive Rules):** Rules which regulate proper interpretation of a given rule or its application.

<sup>34</sup> अणुदित्सवर्णस्य चाप्रत्ययः

<sup>35</sup> Shastri, Bheemsen, Laghusiddhaantkaumudi Ist part, page: 5

<sup>36</sup> संज्ञा च परिभाषा च विधिर्नियम एव च ।  
अतिदेशोऽधिकारश्च षड्विधं सूत्रमुच्यते ॥

- **Vidhi (Operational Rules):** Rules which state a given operation to be performed on a given input.
- **Niyama (Restriction Rules):** Rules which restrict the scope of a given rules.
- **Atideśa (Extensions Rules):** Rules which expand the scope of a given rules, usually by allowing the transfer of certain properties which were otherwise not available.
- **Adhikāra (Heading Rules):** Rules which introduce a domain of rules sharing a common topic, operation, input, physical arrangement, etc.

### ***Samjñā* (introduce classes and technical rules)**

Pāṇini's *samjñā* rules define terms or conventions (or macros) to be used in the grammar. The object which is assigned a *samjñā* is called a *samjñī*. Pāṇini has used approximately 100 technical terms which can be classified in view of their *samjñā* in three categories

- *śabdasamjñā* assigns a *samjñā* to a linguistic term, for example, *vṛddhi*, *prātipadika*, *dhātu*, *aṅga*, *pada* etc.
- *arthasamjñā* assigns a *samjñā* to the meaning of linguistic term, for example, *luk*, *lopa*, *ślu*, *lup* etc.
- *dharmasamjñā* assigns a *samjñā* to the quality (*guṇa*) of a sound segment, for example, *adarśana*, *udātta*, *anudātta svarita* etc.

### ***Paribhāṣā* (Interpretive Rules)**

The interpretive rules called *paribhāṣā* provide a check on the operational rules so that they do not suffer from faults such as *ativyāpti* (over application), *avyāpti* (under application) and *asambhava* (impossible application). Pāṇini has located about seventy-five such interpretive rules in different parts of the grammar. For example, *ādyantau*

*ṭakitau* [1.1.47], *midaco'ntyādparaḥ* [1.1.47], *ṣaṣṭhī sthāne yogā*[1.1.49], *sthāne'ntaratamaḥ* [1.1.50], *uraṇ raparaḥ* [1.1.51], *nicca*<sup>37</sup> [1.1.53] etc.

### **Vidhi (Operational)**

Operation rules provide for a certain operation to be performed. In this sense, the term *vidhi* refer to *kārya* (operation, action) *vidhi* is also used in the sense of the object of an operation. The derivational mechanism of the *Aṣṭādhyāyī* entails the following operation-

- Placement (*pratyaya*) like *pratyayaḥ* [3.1.1], *paraśca* [3.1.2], *dhātoḥ* [3.1.91] etc.
- Addition (*āgama*) like *hrasvanadyāpo nuṭ* [7.1.54], *āmi sarvanāmnaḥ suṭ* [7.1.52] etc
- Replacement (*ādeśa*) like *akaḥ savarṇe dīrghaḥ* [6.1.101], *ecoayavayāvah* [6.1.78], *ikoyaṇaci* [6.1.77], *ato'm* [7.1.24] etc.
- Modification (*vikaraṇa*) like *kartari śap* [3.1.68], *divādibhyaḥ śyan* [3.1.69] etc.

Deletion (*lop*) like *halantyaṃ* [1.3.3], *laśakvataddhite* [1.3.8], *upadeśe'janunāsik it* [1.3.2] etc.

### **Niyama (Restriction)**

Rules which restrict the scope of a given rule.

Examples: *goto nit* [7.1.90], *striyāñca* [7.1.96], *trīyādiṣu bhāṣitapuṃskam puṃvad gālavyasya* [7.1.74]<sup>38</sup> etc.

### **Atideśa (Extensions)**

The function of an extension rule is to widen the scope of application of a technical or operation rule. *atideśa* is of three types

- *samjñātideśa* like *asamyogālliṭ kit* [1.2.5]
- *sthānyātideśa* like *sthānivadādeśo'nalvidhau* [1.1.46]

<sup>37</sup> आदन्तौ टकितौ [१.१.४७], मिदचोऽन्त्यादपरः [१.१.४७], षष्ठी स्थाने योगा [१.१.४९], स्थानेऽन्तरतमः [१.१.५०], उरण् रपरः [१.१.५१], डिच्च [१.१.५३].

<sup>38</sup> गोतो णित् [७.१.९०], स्त्रियाञ्च [७.१.९६], तृतीयादिषु भासितपुंस्कं पुंवद् गालवस्य [७.१.७४]



- *yuktātideśa* like *lupi yuktavad vyaktivacane* [1.2.51]

### ***Adhikāra* (Heading)**

Rules which introduce a domain of rules sharing a common topic, operation, input, physical arrangement, etc.

Examples: *pratyayaḥ* [3.1.1], *paraśca* [3.1.2], *taddhitāḥ* [4.1.76], *dhātoḥ* [3.1.91], *sahasupā* [2.1.4] etc.

### ***Dhātupāṭha* (1967 verb roots - 2014 including *kaṇḍvādi* roots) (DP).**

The *dhātupāṭha* is a lexicon of [Sanskrit](#) verb [roots](#) assumed or explicitly called by the SP component. There are 1967 verb roots, 2014 including *kaṇḍvādi* roots in [Pāṇini](#) *dhātupāṭha*. It is organized into ten classes as follows –

<i>Sr.</i>	<i>Class</i>	<i>Total roots</i>	<i>Modification</i>
1	bhvādi	1035	śap
2	adādi	71	luk
3	juhotyādi	24	śu
4	divādi	141	śyan
5	svādi	34	śnu
6	tudādi	155	śa
7	rudhādi	25	śnam
8	tanādi	10	u
9	kryādi	62	śnā
10	curādi	410	śap
<b>Total</b>	<b>10</b>	<b>1967</b>	<b>10</b>

**Table-1.2 : Distribution of DP**

## ***gaṇapāṭha* (GP)**

The primitive nominal bases are contained in the GP. The various classes like *kṛt*, *taddhita*, *strī*, *sup*, *tin* and the 18 *upasargas* operate on these bases (including 23 pronouns)<sup>39</sup>. The *gaṇapāṭha* is a list of groups of words used by Pāṇini's [Aṣṭādhyāyī](#). For example: *sarvādi*, *ajādi*, *śaradādi* etc.

### **1.2.1.1 Technical terms of Pāṇini**

In addition to *anuvṛtti* and artificial technical terminology including *pratyāhāras*, Pāṇini employs a device called *anubandha*. An *anubandha* is a code-letter which indicates a grammatical function like elision and reduplication. A major aspect of Pāṇini's technique is the law of *utsarga* and *apavāda* that relates exceptions and individual rules. The law of *utsarga* and *apavāda* states that an *apavāda* (exception rule) is more powerful than an *utsarga* (general rule). Therefore before applying the *utsarga* one has to check for its *apavāda*. The *utsarga* thus occupies the area not occupied by its exceptions. Further, once an *utsarga* is barred from entering in to the area of its exception, it can never enter the area again. For example: Pāṇini's rule *ata iñ* [P.4.1.95] (the suffix *iñ* is added to a noun ending in *a* in the sense of offspring), is an exception (*apavāda*) to Pāṇini's rule *tasyāpatyam (aṇ)* [4.1.92] (the suffix *aṇ* is added to a noun in the sense of its offspring). *Siddha*, *asiddha*, *nitya*, *anitya*, *antaraṅga* and *bahiraṅga* are the important technical term of Pāṇini. When the two rules of equal strength (*vipratishedha*) conflict, the one which is subsequent in order, applies. If there is a situation where a single element is equally qualified for the assignment of more than one term, *vipratishedhe param kāryam* [1.4.2] will be invoked. It should be remembered here that unless there is a clear conflict, it should not be invoked. Commentators also explain that this rule cannot be invoked where the conflict obtains between two rules whose relationship is one of general-exception (*utsarga-apavāda*), obligatory-nonobligatory (*nitya-anitya*) or internally conditioned-externally conditioned (*antaraṅga -bahiraṅga*)<sup>40</sup>. The concept of *adhikāra* is intended to regulate the meaning of the rules to follow in the sense that the whole of *adhikāra* rule is to be read with the subsequent *sūtras*. For example, *dhātoḥ* [3.1.91] is an *adhikāra sūtra*, which applies till the end of the third

<sup>39</sup> Jha Girish Nath 'The System of Panini' Language in india <http://www.languageinindia.com/feb2004/panini.html> access on 14th march 2006

<sup>40</sup> उत्सर्गापवादमित्यादित्यान्तरङ्गबहिरङ्गेषु तुल्यबलता नास्ति [काशिका-१.४९९]

chapter. Anything treated after this rule will get the designation *dhātoḥ*. Pāṇini's *saṃjñā sūtra* introduces different classes and abbreviations that are to be called in the *sūtras* – *vr̥ddhi*, *guṇa*, *anunāsika*, *savarṇa*, *prātipadika*, *pada*, *sarvanāmasthāna*, *ghi*, *ghu*, *gha*, *nadī*, *bha*, *niṣṭhā*, *ṭi*, *pragrhya*, *hrasva-dīrgha-plut*, *kṛtya* etc<sup>41</sup>. *saṃjñā* is the most important technical device of Pāṇini.

## 1.2.2 Nominal Inflectional Morphology (*subanta*) of Pāṇini

In a Sanskrit sentence, all non-verb categories are *subanta-padas* which makes it essential to analyze these *padas* before any other computer processing can begin. Sanskrit *subanta* forms can be potentially very complex. They can include primary (*kṛdanta*) and secondary (*taddhitānta*), feminine forms (*strīpratyayānta*) and compound nouns (*samāsa*). They can also include *upasargas* and *avyayas* etc. According to Pāṇini, there are 21 morphological suffixes (seven *vibhaktis* and combination of three numbers = 21)<sup>42</sup> which can attach to the nominal bases (*prātipadika*) according to the syntactic category of the base, gender and end character of the base. Pāṇini has listed the *sup* suffixes *su*, *au*, *jas*, *am*, *auṣ*, *śas*, *tā*, *bhyām*, *bhis*, *ñe*, *bhyām*, *bhyas*, *ñasi*, *bhyām*, *bhyas*, *ñas*, *os*, *ām*, *ni*, *os*, *sup*. These suffixes are in the sets of three as- (*su*, *au*, *jas*) (*am*, *auṣ*, *śas*) (*tā*, *bhyām*, *bhis*) (*ñe*, *bhyām*, *bhyas*) (*ñasi*, *bhyām*, *bhyas*) (*ñas*, *os*, *ām*) (*ni*, *os*, *sup*)<sup>43</sup> for singular, dual and plural<sup>44</sup> respectively. These suffixes are added to the *prātipadikas*<sup>45</sup> (any meaningful form of a word, which is neither a root nor a suffix) to obtain inflected forms (*subanta padas*). *prātipadikas* are of two types: primitive and derived. The primitive bases are stored in *gaṇapāṭha* (collection of bases with similar forms) while the latter are formed by adding the derivational suffixes. They denote unity, duality and plurality respectively. Some words are only in the singular always, like *ekaḥ*(one), some are always dual like *dvi* (two), *akṣi* (eyes) etc. and some are always plural like *apah* (water), *dārāḥ* (wife) etc. *subanta* is mainly six types -

<sup>41</sup> बुद्धिसदैच [१.१.१], अदेङ्गुणः [१.१.२], मुखनासिकावचनोऽनुनासिकः [१.१.८], तुल्यास्यप्रयत्नं सवर्णम् [१.१.९], सुप्तिङन्तं पदम् [], सुङ् नपुंसकस्य [१.१.४३], अर्थवधातुरप्रत्ययः प्रातिपदिकम् [१.२.४५] कृत्तद्धितसमासाश्च [१.२.४६], शेषोऽध्यसखि [१.४.७], दाधाध्वदाप् [१.१.२०], तरप्तमपौ घः [१.१.२२], यूस्त्र्याख्यौ नदी [१.४.३], यचिभम् [१.४.१८], क्तक्तवतू निष्ठा [१.१.२६], अचोऽन्त्यादि टि [१.१.६४], ईहेद् द्विवचनं प्रगृह्यम् [१.१.११], ऊकालोऽज्झस्व-दीर्घ-प्लुतः [१.३.२७], कृत्याः [३.१.११] आदि ।

<sup>42</sup> स्वौजसमौट्छष्टाभ्याम्भिस्ङेभ्याम्भ्यस्ङसिभ्याम्भ्यस्ङसोसांङ्योस्सुप्

<sup>43</sup> सुपः

<sup>44</sup> द्व्येकयोर्दिवचनैकवचने

<sup>45</sup> अर्थवधातुरप्रत्ययः प्रातिपदिकम् ।१।२।४५॥, कृत्तद्धितसमासाश्च ।१।२।४६॥

### 1.2.2.1 *avyaya subanta* ( indeclinable NPs)

*avyaya subanta-padas* remain unchanged under all morphological conditions<sup>46</sup>. According to Pāṇini [2.2.82]<sup>47</sup>, affixes *cāp*, *ṭāp*, *ḍāp*, (feminine suffixes) and *sup* are deleted by *luk* when they occur after an *avyaya*. Pāṇini defines *avyayas* as *svarādinipātamavyayam* [1.1.36], *kṛnmejantaḥ* [1.1.38], *ktvā tosun kasunaḥ* [1.139] and *avyayībhāvaśca* [1.1.40]<sup>48</sup> etc.

### 1.2.2.2 Nominal *Subanta* (base NPs)

Nominal *subantas* are basic *subanta*, which are *prātipadika* by *arthavadadhāturapratyayaḥ prātipadiakam*. For example: *rāmaḥ*, *śyāmaḥ*, *pustakālayaḥ*, *vidyālayaḥ* etc.

### 1.2.2.3 *samāsānta subanta* (compound NPs)

Simple words (*padas*), whether substantives, adjectives, verbs or indeclinables, when added with another *subanta pada* form *samāsa* (compound). Sanskrit *samāsas* are divided into four categories, some of which are divided into sub-categories. The four main categories of compounds are as follows: (1) Adverbial or *avyayībhāva*, (2) Determinative or *tatpuruṣa*, (3) Attributive or *bahuvrīhi* and (4) Copulative or *dvandva*. *dvandva* and *tatpuruṣa* compounds may be divided into sub-categories also.

### 1.2.2.4 *kṛdanta subanta* (primary derived NPs)

The primary derivatives are called *kṛdanta*. The primary affixes are to be added to verbs to derive substantives, adjectives or indeclinable *kṛt*. For example *paṭhitavyam*, *pātavya*, *paṭhanīya*, *pacelima*, *jeyam*, *deyam*, *karttā*, *kumbhakāraḥ*, *janamejayaḥ*, *pāṭhakaḥ*, *paṭhantī*, *gantum*, *khāditum*, *svapnam gatiḥ*, *gatvā*, *vihāya*, *ādāya* etc.

<sup>46</sup> सदृशं त्रिषु लिङ्गेषु सर्वासु च विभक्तिषु ।

वचनेषु च सर्वेषु यन्न व्येति तदव्ययम् ॥ [गोपथ ब्राह्मण]

<sup>47</sup> अव्ययादाप्सुपः [२.४.८२]

<sup>48</sup> स्वरादिनिपातमव्ययम् [१.१.३६], कृन्मेजन्तः [१.१.३८], क्त्वा-तुसुन्-कसुनः [१.१.३९], अव्ययीभावश्च [१.१.४०]

### 1.2.2.5 *taddhitānta subanta* (secondary derived NPs)

The secondary derivative affixes are called *taddhita*, which derive secondary nouns from *prātipadikas*. For example - *dāśarathī*, *gauṇa* etc. Pāṇini has listed many *taddhita* suffixes some of which are- *a*, *akañc*, *ac*, *añ*, *aṇ*, *at*, *iṣṭhan*, *īyasun*, *kan*, *ḍhak*, *ḍhañ*, *tamap*, *tarap*, *tayap*, *tal*, *tyap*, *tral*, *dvayasac*, *fak*, *matup*, *mātrac*, *yak*, *yat*, *yañ*, *ḍāc*, *kha*, *gha*, *cha*, *uraca*, *ṭhak*, *ṭhañ*, *ṭhan*, *na*, *ha*, *va*, *vatup* etc. For example, *dākṣī*, *kva*, *aśvakaḥ*, *viśvajanīnam*, *kṣatriyaḥ*, *mālīyaḥ*, *raivatikaḥ*, *dāṇḍikaḥ*, *laghutamaḥ*, *gurutaraḥ*, *gārgyāyaṇaḥ*, *iha*, *balavān* etc.

### 1.2.2.6 *strīpratyayānta subanta* (feminine derived NPs)

Sanskrit has eight feminine suffixes *īāp*, *cāp*, *ḍāp*, *nīṣ*, *nīn*, *nīp*, *un* and *ti* etc. and the words ending in these suffixes are called *strīpratyayānta*. For example - *ajā*, *gaurī*, *mūṣikā*, *indrāṇī*, *gopī*, *aṣṭādhyāyī*, *kurucarī*, *yuvatī*, *karabhorū* etc.

## **Chapter- II**

### **SUBANTA TYPES AND PROCESSES: THEIR RECOGNITION**

## 2.1 Constituents of Sanskrit sentence

A sentence in Sanskrit has *subanta padas* (NPs) and *tinanta padas* (VPs). **Cordona<sup>49</sup> (1988)** gives his reference as footnote) defines a sentence as -

$$(N - E^n)p \dots (V - E^v)p$$

After *sup* and *tin* combine with *prātipadikas*, they are assigned syntactico-semantic relation by the *kāraka* stipulations to return complete sentences

### 2.1.1 Subanta and its morphological types

Nominal inflection morphology (*subanta*) deals with combination of bases (*prātipadika* (*arthavadadhāturapratyayaḥ prātipadiakam*)) with case affixes (*sup*). The words (*padas*) thus formed are called *subanta*. For example, *rāmaḥ*, *śyāmaḥ*, *pustakālayaḥ*, *vidyālayaḥ* etc. *subantas* are of two types -

#### 2.1.1.1 Primary subanta

The primary derivatives are called *kṛdanta*. The *kṛt* affixes are also known as primary affixes. The primary affixes are to be added to verbs. When added to the verbs, the *kṛt* can derive substantives, adjectives and indeclinables. The Sanskrit names for these two types of bases are as follows: *kṛdanta* (a word ending in a *kṛt* affix) and *taddhitānta* (a word ending in a *taddhita* affix). *kṛt* suffixes are of three kinds according to the *Siddhāntakaumudī* (SK) by *Bhaṭṭojidīkṣita-*

##### 1. kṛtya suffixes

*kṛtya* suffixes are always used in *bhāva-vācya* and *karma-vācya* and are in neuter singular. *kṛtya* suffixes are *tavyat*, *tavya*, *anīyar*, *kelimar*, *yat*, *kyap*, *nyat* etc<sup>50</sup>. For example- *paṭhitavyam*, *pātavya*, *paṭhanīya*, *pacelima*, *jeyam*, *deyam*, etc.

<sup>49</sup> **George Cardona, 1988** Pāṇini, His Work and its Traditions, vol ... i (Delhi: **MLBD**, 1988)

<sup>50</sup> तव्यं तव्यतञ्चेवाऽनीयर् केलिमरौ तथा ।  
यत् प्यत् क्यप् चापि कृत्यान् सप्त प्रचक्षते ॥

## 2. *pūrvakṛdanta* suffixes

These suffixes are used in *karṭṛvācya* only. These suffixes are *ṇvul, tṛc, lyu, ṇini, ac, ka, aṇ, ṭa, khaś, khac, kvanip, ḍa, kta, ktavatu, śatr, śānac, śākan, u, kvip, itra, etc.* For example *karṭtā, kumbhakāraḥ, janamejayah, pāṭhakaḥ, paṭhantī* etc.

## 3. *uttarakṛdanta* suffixes

These suffixes are *tumun, ghañ, erac, ap, kṛ, athuc, nañ, nan, ktin, khal, yuc, ktvā, lyap, ṇamula, uṇa* etc. For example, *gantum, khāditum, svapnam, gatiḥ, gatvā, vihāya, ādāya* etc. Primary derived nouns (by way of *kṛt* suffixation) get inflected for case and number by sup suffixes to become *subanta-padas* (*kṛdanta subanta-padas*).

### 2.1.1.2 Secondary *subanta*

Nouns formed by secondary derivation process are called secondary *subanta*, for example, *taddhitānta, samāsanta, strīpratyayānta* and *avyayas*.

#### 2.1.1.2.1 *taddhitānta subanta*

The secondary derivatives are called *taddhita*. *taddhitānta* are words ending in *taddhita* affixes before getting sup inflections. *taddhita* affixes derive secondary nouns and change their meanings in various ways for example - *dāśarathī, gaṇ* etc. Pāṇini described many *taddhita* suffixes. Some suffixes are- *a, akañc, ac, aṇ, aṇ, at, iṣṭhan, īyasun, kan, ḍhak, ḍhañ, tamap, tarap, tayap, tal, tyap, tral, dvayasac, fak, matup, mātrac, yak, yat, yañ, ḍāc, kha, gha, cha, uraca, ṭhak, ṭhañ, ṭhan, na, ha, va, vatup* etc. For example - *dākṣī, kva, aśvakaḥ, viśvajanīnam, kṣatriyaḥ, mālīyaḥ, raivatikaḥ, daṇḍikaḥ, laghutamaḥ, gurutarah, gārgyāyaṇaḥ, iha, balavān* etc. *taddhitānta* forms inflected with *sup* are called *taddhita-subanta*.

#### 2.1.1.2.2 *samāsānta subanta*

Simple words (*padas*), whether substantives, adjectives, verbs or indeclinable, added with another *subanta-padas* are called *samāsa* (compound). Sanskrit *samāsas* are divided into four categories, some of which are in turn divided into sub-categories. The four main categories of



compounds are as follows: (1) adverbial or *avyayībhāva*, (2) determinative or *tatpuruṣa*, (3) attributive or *bahuvrīhi* and (4) copulative or *dvandva*. *dvandva* and *tatpuruṣa* compounds can further be subdivided into sub-categories.

### 2.1.1.2.3 *strīpratyayānta subanta*

Sanskrit has eight feminine suffixes *ṭāp*, *cāp ḍāp*, *nīs*, *nīn nīp*, *unī* and *ti* and words ending in these suffixes are called *strīpratyayānta*. For example - *ajā*, *gaurī*, *mūṣikā*, *indrāṇī*, *gopī*, *aṣṭādhyāyī*, *kurucarī*, *yuvatī*, *karabhorū* etc. *strīpratyayānta* forms inflected with *sup* are called *strīpratyayānta-subanta*.

### 2.1.1.2.4 *avyaya subanta*

Forms remaining unchanged in all genders, numbers and cases, are called *avyaya*.<sup>51</sup> *avyayas* are basically *subanta-pada* but according to Pāṇini's rule *avyayādāp supaḥ* [2.2.82]<sup>52</sup> feminine affixes *cāp*, *ṭāp*, *ḍāp*, and *sup* are deleted by *luk* when they occur after an *avyaya*. Pāṇini defines *avyayas* as *svarādinipātamavyayam* [1.1.36], *kṛnmejantaḥ* [1.1.38], *ktvā tosun* [1.1.39] and *avyayībhāvaśca* [1.1.40]<sup>53</sup> etc.

## 2.1.2 *tiṇanta* and its morphological types

Sanskrit verb forms are very complex. They carry tense, aspect, number information in the inflection forms. Sanskrit has about 2000 verb roots classified in 10 morphological and semantic classes. Further, these can have *ātmanepadī* and *parasmaipadī* forms in 10 *lakāra* and 3 x 3 person and number combinations. There are 12 secondary suffixes added to verb roots to create new verb roots<sup>54</sup>. Mishra & Jha (2004)<sup>55</sup> have done a rough calculation of all potential verb forms in Sanskrit to be more than 10,29,60,000.

<sup>51</sup> सदृशं त्रिषु लिङ्गेषु सर्वासु च विभक्तिषु ।

वचनेषु च सर्वेषु यत्र व्येति तदव्ययम् ॥ [गोपथ ब्राह्मण]

<sup>52</sup> अव्ययादाप्सुपः [२.४.८२]

<sup>53</sup> स्वरादिनिपातमव्ययम् [१.१.३६], कृन्मेजन्तः [१.१.३८], क्त्वा-तुसुन्-कसुनः [१.१.३९], अव्ययीभावश्च [१.१.४०]

<sup>54</sup> सन्-क्यच्-काम्यच्-क्यङ्-क्यषोऽथाऽऽचार-क्विब्-णिज्यङ्स्तथा ।

यगाय ईयङ् णिङ् चेति द्वादशाऽमी सनादयः ॥

### 2.1.2.1 Primary *tiñanta*

Primary *tiñanta* forms are derived by directly inflecting the verbs with one of the following 18 suffixes called *tiñ* (*tip, tas, jhi, sip, thas, tha, mip, vas, mas, ta, ātām, jha, tha, āthām, dhvam, iṭ, vahin, mahin*). For example, *bhavati, paṭhati, gacchati, bhramati, krāmati, edhate, dadhate* etc.

### 2.1.2.2 Derived *tiñanta*

There are 12 secondary suffixes (*san, kyac kāmīyac, kvip, kyañ, kyaṣ, kvib, ñic, yañ, yak, ṛyañ* and *ñin*.) which derive new verb roots from existing verb roots. The verb roots thus derived are divided into four classes -

1. causals or *ñijantas* like *corayati, pāṭhayati, gamayati, darśayati* etc.,
2. desiderative or *sannantas* like *pipaṭhiṣati, mumūrṣati, jugupsīṣate, vivṛtsati, cikīrṣati* etc,
3. frequentatives or *yañantas* like *lolupyate, sāsadyate, cañcūryate, cekrīyate, vāvrajjyate* etc. and
4. denominative or *nāmadhātus* - *kyac* (e.g. *purīyati, kavīyati, rājīyati*), *kyañ* (e.g. *kṛṣṇāyate, vidvāyate, yaśasyate, apsarāyate*), *kyaṣ* (e.g. *paṭaptāyate, lohītāyati*), *ñin* (e.g. *sambhāṇḍayate, muṇḍayati*), *yak* (e.g. *sarpayati, kaṇḍūyati, kelāyati*).

## 2.2 *subanta* formation processes

A *pada* is defined as *suptiñantam padam*. A *subanta-pada* (NP) is a combination of nominal *prātipadika* - base or derived (formed by *kṛt, taddhita, strī* or *samāsa* process) - with one of the 21 *sup* suffixes depending on case number combinations. After a *sup* suffix is selected (depending on case-number combination), the *prātipadika-sup* structure changes according to the following information –

- category of the *prātipadika* (nouns, pronouns, numerals)
- gender of the *prātipadika* (masculine, feminine, neuter)
- last *varṇa* (ending character) of the *prātipadika* (vowels/consonants)

---

<sup>55</sup> Mishra, Sudhir and Jha, Girish Nath, identifying verb inflection in Sanskrit Morphology

## 2.2.1 Vowel ending *prātipadika*

### ‘अ’ ending masculine

For nominative singular (1-1), suffix is ‘*su*’ [P.1.3.2]<sup>56</sup>, which is replaced by ‘*visarga*’ (:). There are five steps in replacing ‘*visarga*’ (*su* > *s* > *ru* > *r* > :). For example- *rāmaḥ*, *śyāmaḥ*, *sarvaḥ* etc (रामः, श्यामः, सर्वः). Accusative singular (2-1) ‘*am*’ changes to ‘*m*’ [6.1.106]<sup>57</sup> blocking *dīrgha sandhi* and generates forms *rāmam*, *śyāmam* etc. For instrumental singular (3-1), the suffix is ‘*ṭā*’, which is replaced by ‘*in*’ [7.1.12]<sup>58</sup> then by applying the rule for *guṇa sandhi* [6.1.87]<sup>59</sup> which replaces ‘*ai*’ *saṃhitā* by ‘*e*’ ( *ṭa*>*ina*>*ena*>*eṇa*) leading to forms like *rāmeṇa*, *śyāmeṇa* etc. For dative singular (4-1), the suffix is ‘*ṇe*’ (*e*), which changes to ‘*aya*’ [1.3.8]<sup>60</sup> thus resulting forms like *rāmāya*, *śyāmāya* etc. For ablative singular (5-1), the suffix is ‘*ṇasi*’, which is replaced by ‘*āt*’ [7.1.12]<sup>61</sup>. Then by applying *dīrgha sandhi* rule [6.1.101]<sup>62</sup> we get forms like *rāmāt*, *śyāmāt* etc. For genitive singular (6-1), the suffix is ‘*ṇas*’ which is replaced by ‘*sya*’ [7.1.12]<sup>63</sup> to get forms like *rāmasya*, *śyāmasya* etc. For locative singular (7-1), the suffix is ‘*ṇi*’ which is replaced by ‘*i*’ and then by applying rule [6.1.87]<sup>64</sup> we get forms like *rāme*. For vocative, ‘*he*’ is prefixed to *visarga* less nominative singular to get forms like ‘*he rāma*’ (*he rāma* + *su*[2.3.49]<sup>65</sup> = ‘*he rāma*’ [6.1.69]<sup>66</sup>)

For nominative and accusative dual, suffixes are *au* and *auṭ*. They change to *au* [P.1.3.2]<sup>67</sup>. Then the *sandhi* rule<sup>68</sup> applies. The resulting forms are like *rāmau*, *śyāmau*, *bālakau*, *bharatau* etc. For instrumental, dative and ablative dual (3-2/4-2/5-2), the suffix is *bhyām*, the

---

<sup>56</sup> उपदेशेऽजनुनासिक इत्

<sup>57</sup> अमिपूर्वः

<sup>58</sup> टाडसिडसामिनात्स्याः

<sup>59</sup> आद् गुणः

<sup>60</sup> लशक्वतद्धिते

<sup>61</sup> टाडसिडसामिनात्स्याः

<sup>62</sup> अकः सवर्णे दीर्घः

<sup>63</sup> टाडसिडसामिनात्स्याः

<sup>64</sup> आद् गुणः

<sup>65</sup> एकवचनम् सम्बुद्धिः

<sup>66</sup> एङ्-ह्रस्वात् संबुद्धेः

<sup>67</sup> उपदेशेऽजनुनासिक इत्

<sup>68</sup> चुटू

last character of base changes to ‘ā’ by the rule [7.3.102]<sup>69</sup> and there is no change in suffix. For example: *rāmābhyām*, *bharatābhyām*, *śyāmābhyām* etc. For genitive and locative dual (6-2/7-2), the suffix is ‘os’ which changes to ‘oḥ’ and the last character of base changes to ‘e’ by the rule [7.3.104]<sup>70</sup> and applies *sandhi* rule [6.1.78]<sup>71</sup>. The resulting forms are like *rāmayoḥ*, *śyāmayoḥ*, *bharatayoḥ* etc. The vocative dual (8-2) forms are like nominative dual with prefix ‘he’ like ‘he *rāmau*’.

For nominative plural (1-3), the suffix is ‘jas’. It converts to ‘as’ and by applying *sandhi* rule [1.3.7, 6.1.101]<sup>72</sup> generates *rāmāḥ*, *śyāmāḥ*, *bharatāḥ*. For accusative plural (2-3), the suffix is ‘sas’ → ‘as’ by the rule 1.3.8<sup>73</sup> which changes to ‘ān’ [6.1.103]<sup>74</sup>. Then by applying *sandhi* rule [6.1.101]<sup>75</sup> we get forms like *rāmān*, *śyāmān*, *bharatān* etc. For instrumental plural (3-3), the suffix is ‘bhis’ → ‘ais’. The forms are like *rāmaiḥ*, *śyāmaiḥ*, *bharataiḥ* etc. by applying ‘*vṛddhi sandhi*’ [6.1.88]<sup>76</sup>. For dative and ablative plural (4-3/5-3), the suffix is ‘bhyas’ which changes the last ‘a’ of the base to ‘e’ by the rule 7.3.103<sup>77</sup>. For example: *rāmebhyaḥ*, *śyāmebhyaḥ*, *bharatebhyaḥ* etc. For genitive plural (6-3), suffix is ‘ām’ and gets an augment ‘nuḥ’ by the rule 7.1.54<sup>78</sup> which changes to ‘n’ [1.3.3]<sup>79</sup> and combines with ‘ām’ before changing the last character of the base to ‘ā’ by the rule 6.4.3<sup>80</sup>. n → ṇ conversion takes place by the rule 8.4.2<sup>81</sup>. For example - *rāmāṇām*, *bharatāṇām*, *śyāmāṇām* etc. For the locative plural (7-3), the suffix is ‘sup → su’. The final last character of the ‘a’ ending base is converted to ‘e’ by the rule 7.3.103<sup>82</sup>. The suffix ‘su’ changes to ‘ṣu’ by 8.3.59<sup>83</sup>. For example- *rāmeṣu*, *śyāmeṣu*, *bharateṣu* etc.

<sup>69</sup> सुपि च

<sup>70</sup> ओसि च

<sup>71</sup> एचोऽयवायावः

<sup>72</sup> चुटू, अकः सवर्णे दीर्घः

<sup>73</sup> लशक्वतद्धिते

<sup>74</sup> तस्माच्छसो नः पुंसि

<sup>75</sup> चुटू, अकः सवर्णे दीर्घः

<sup>76</sup> वृद्धिरेचि

<sup>77</sup> बहुवचने झल्येत्

<sup>78</sup> ह्रस्व नद्यापो नुट्

<sup>79</sup> हलन्त्यम्

<sup>80</sup> नामि

<sup>81</sup> अट्कुप्वाङ्नुम्व्यवाऽयेपि

<sup>82</sup> बहुवचने झल्येत्

<sup>83</sup> आदेश प्रत्यययोः

There are some differences between ‘a’ ending masculine pronouns and nouns in some cases. Sanskrit has 35 pronouns according to Pāṇinī’s rule of ‘sarvaṇāma’ [1.1.27]<sup>84</sup>. Except in the case of [(1-3), (4-1), (5-1), (6-3) and (7-1)], all the forms of ‘a’ ending masculine pronouns are same as that of ‘a’ ending masculine nouns. So we will describe here in the short only ‘sarva’ (S) which is representative of the class ‘a’ ending masculine pronouns generated following process:

- 1-3 : S + jas → S + śī → S + ī → sarve [7.1.17/1.3.8/6.1.87]<sup>85</sup>  
 4-1 : S + ñe → S + smai [7.1.14]<sup>86</sup> → sarvasmai  
 5-1 : S + ñasi → S + smāt [7.1.15]<sup>87</sup> → sarvasmāt  
 6-3 : S + ām → S + suṭ + ām → S + s + ām → S + sām →  
 sarve + sām → sarve + ṣām → sarveṣām [7.1.52/1.3.3/7.3.103/8.3.59]<sup>88</sup>  
 7-1 : S + ñi → S + smin [7.1.15]<sup>89</sup> → sarvasmin

### ‘अ’ ending neuter

Except nominative, accusative and vocative forms [(1-1), (1-2), (1-3), (2-1), (2-2), (2-3), (8-1), (8-2) and (8-3)] all the forms of ‘a’ ending neuter nouns are same as that of ‘a’ ending masculine nouns. So we take ‘jñāna’ (J) ‘a’ ending neuter noun:

- 1-1 : J + su → J + am → jñānam [7.1.24/6.1.103]<sup>90</sup>  
 1-2 : J + au → J + śī → J + ī → jñāne [7.1.19/1.3.8/ 6.1.84]<sup>91</sup>  
 1-3 : J + jas → J + śi → J + i → J + num + i →  
 J + n + i → jñānāni [7.1.20/1.3.8/ 7.1.72/1.3.3/6.4.8]<sup>92</sup>  
 2-1 : J + am → jñānam [6.1.103]<sup>93</sup>

<sup>84</sup> सर्वादीनि सर्वनामानि

<sup>85</sup> जसः शी / लशक्वतद्धित / आद् गुणः

<sup>86</sup> सर्वनाम्नः सर्वस्मै

<sup>87</sup> डसि-डयोः-स्मात्-स्मिनौ

<sup>88</sup> आमि सर्वनाम्नः सुट् / हलन्त्यम् / बहुवचने झल्येत / आदेश प्रत्यययोः

<sup>89</sup> डसि-डयोः-स्मात्-स्मिनौ

<sup>90</sup> अतोऽम् / तस्माच्छसो नः पुंसि

<sup>91</sup> औङ् आपः / लशक्वतद्धिते / एकः पूर्वपरयोः

<sup>92</sup> जस् शसोः शिः / लशक्वतद्धिते/ नपुंसकस्य झलचः / हलन्त्यम् / सर्वनाम्स्थाने चासंबुद्धौ

<sup>93</sup> तस्माच्छसो नः पुंसि

- 2-2 : J + auṭ → J + śī → J + ī → jñāne [7.1.19/1.3.8/6.1.84]<sup>94</sup>
- 2-3 : J + jas → J + śi → J + i → J + num + i [7.1.20/1.3.8/7.1.72]<sup>95</sup> →  
J + n + i → jñānāni [1.3.3/6.4.8]<sup>96</sup>

### ‘आ’ ending masculine

This is smaller class comparatively other classes the processing of ‘viśvapā’ (V) is representative of this class:

- 2-3 : V + śas → viśvapā + as [6.4.140/1.1.52/]<sup>97</sup> → viśvapāḥ
- 3-1 : V + ṭā → viśvap + ā [6.4.140/1.1.52/1.3.7]<sup>98</sup> → viśvapā
- 3-3 : V + bhis → viśvapābhiḥ
- 4-1 : V + ñe → viśvapa + e [6.4.140/1.1.52/1.3.8]<sup>99</sup> → viāvape
- 5-1 : V + nasi → viśvap + as [6.4.140/1.1.52/1.3.8]<sup>100</sup> → viśvapāḥ
- 6-1 : V + nas → viśvap + as [6.4.140/1.1.52/1.3.8]<sup>101</sup> → viśvapāḥ
- 6-2/7-2 : V + os → viśvap + oḥ [6.4.140/1.1.52/1.3.8]<sup>102</sup> → viśvapoh
- 6-3 : V + ām → viśvap + ām [6.4.140/1.1.52]<sup>103</sup> → viśvapām
- 7-1 : V + ni → viśvap + i [6.4.140/1.1.52/1.3.8]<sup>104</sup> → viśvapi

Except these forms, all other forms are like ‘ā’ ending masculine noun bases. If a non ‘bha’ base does not end in a root (for example ‘viśvapā’) then its forms will be like ‘hā’ as follows <sup>105</sup> -

- 2-3 : H + śas → H + an [6.1.103]<sup>106</sup> → hāhān
- 3-1 : H + ṭā → H + ā → hāhā [1.3.8/6.1.101]<sup>107</sup>

<sup>94</sup> लशक्वतद्धित / नपुंकाच्च / एकः पूर्वपरयोः

<sup>95</sup> लशक्वतद्धिते / जस् शसोः शिः / नपुंसकस्य झलचः

<sup>96</sup> हलन्त्यम् / सर्वनाम्स्थाने चासंबुद्धौ

<sup>97</sup> आतो धातोः, अलोऽन्त्यस्य

<sup>98</sup> आतो धातोः, अलोऽन्त्यस्य, चुटू

<sup>99</sup> आतो धातोः, अलोऽन्त्यस्य, लशक्वतद्धिते

<sup>100</sup> आतो धातोः, अलोऽन्त्यस्य, लशक्वतद्धिते

<sup>101</sup> आतो धातोः, अलोऽन्त्यस्य, लशक्वतद्धिते

<sup>102</sup> आतो धातोः, अलोऽन्त्यस्य, लशक्वतद्धिते

<sup>103</sup> आतो धातोः, अलोऽन्त्यस्य

<sup>104</sup> आतो धातोः, अलोऽन्त्यस्य, लशक्वतद्धिते

<sup>105</sup> यचिभम्

<sup>106</sup> तस्माच्छसो नः पुंसि

- 4-1 : H + ñe → H + e → hāhai [1.3.8/6.1.88]<sup>108</sup>  
 5-1 : H + ñasi → H + as → hāhāḥ [1.3.8/1.3.2/6.1.101]<sup>109</sup>  
 6-1 : H + ñas → H + as [1.3.8]<sup>110</sup> → hāhāḥ [1.3.8/6.1.101]<sup>111</sup>  
 6-2/7-2 : H + os → H + oḥ [6.1.88]<sup>112</sup> → hāhauḥ  
 7-1 : H + ñi → H + i → hāhe [1.3.8/6.1.87]<sup>113</sup>

‘आ’ ending feminine:

The ‘ā’ ending feminine form is a very large class. The declension process of this class is represented by ‘ramā’ (R) as follows-

- 1-1 : R + su → R [6.1.88]<sup>114</sup> → ramā  
 1-2/2-2 : R + au/auḥ → R + śī → R + ī → rame [7.1.18/1.3.8/6.1.87]<sup>115</sup>  
 1-3/2-3 : R + jas/śas → R + as → ramāḥ [6.1.101]<sup>116</sup>  
 2-1 : R + am → ramām [6.1.68]<sup>117</sup>  
 3-1 : R + tā → R + ā → rame + ā → ramayā [1.3.7/7.3.105/6.1.78]<sup>118</sup>  
 3-2/4-2/5-2 : R + bhyām → ramābhyām  
 3-3 : R + bhis → ramābhiḥ  
 4-1 : R + ñe → R + e → ramāyā + e → ramāyai [1.3.8/7.3.113/6.1.88]<sup>119</sup>  
 4-3/5-3 : R + bhyas → ramābhyah  
 5-1/6-1 : R + ñasi/ñas → R + as → ramāyāḥ [6.1.68]<sup>120</sup>  
 6-2/7-2 : R + os → rame + os → ramayoḥ [7.3.105/6.1.78]<sup>121</sup>

<sup>107</sup> लशक्वतद्धिते / अकः सवर्णे दीर्घः

<sup>108</sup> लशक्वतद्धिते / वृद्धिरेचि

<sup>109</sup> लशक्वतद्धिते / उपदेशेऽजनुनासिक इत् / अकः सवर्णे दीर्घः

<sup>110</sup> लशक्वतद्धिते

<sup>111</sup> लशक्वतद्धिते / अकः सवर्णे दीर्घः

<sup>112</sup> वृद्धिरेचि

<sup>113</sup> लशक्वतद्धिते / आद् गुणः

<sup>114</sup> वृद्धिरेचि

<sup>115</sup> औङ् आपः / लशक्वतद्धिते / आद् गुणः

<sup>116</sup> अकः सवर्णे दीर्घः

<sup>117</sup> हल्-ङ्याभ्यो दीर्घात् सुतिस्स्यपृक्तं हल्

<sup>118</sup> चुटू / आङि चापः / एचोऽयवायावः

<sup>119</sup> लशक्वतद्धिते / याडापः / वृद्धिरेचि

<sup>120</sup> हल्-ङ्याभ्यो दीर्घात् सुतिस्स्यपृक्तं हल्

- 6-3 : R + aam → R+ naam → ramaaNaam [7.1.54/8.4.2]<sup>122</sup>  
 7-1 : R + ñi → ramāya+ ām → ramāyām [7.3.113/7.3.116/6.1.68]<sup>123</sup>  
 7-3 : R + sup → R+ su → ramāsu  
 8-1 : he R + su → he rame [6.1.69]<sup>124</sup>  
 8-3/8-2 : like 1-2 and 1-3

Except in the cases of [(4-1), (5-1), (6-1), (6-3) and (7-1)], all the forms of ‘ā’ ending feminine pronouns are same as that of ‘ā’ ending feminine nouns. ‘sarbā’ (S) which is representative of the class ‘ā’ ending feminine pronouns has forms generated according to the following process:

- 4-1 : S + ñe → sarvā + e → sarvasyā + e → sarvasyai  
 [1.3.8/7.3.114/6.1.88]<sup>125</sup>  
 5-1/6-1 : S + ñasi/ñas → sarvasyā + as → sarvasyāḥ [7.3.114/1.3.8/6.1.88]<sup>126</sup>  
 6-3 : S + ām → S + s + ām [7.8.52]<sup>127</sup> → sarvāsām  
 7-1 : S + ñi → sarvasyā + ām → sarvasyām [7.3.114/7.3.116/6.1.88]<sup>128</sup>

Other forms are like ‘ramā’.

### ‘इ’ and ‘उ’ ending masculine

‘i’ and ‘u’ ending masculine are derived in the same way because they both are ‘ghi’<sup>129</sup>. The processing of ‘hari’ (H) is as follows-

- 1-1 : H + su → hariḥ  
 1-2/2-2 : H + au/auḥ → harī [6.1.102]<sup>130</sup>

<sup>121</sup> आङि चापः / एचोऽयवायावः

<sup>122</sup> ह्रस्व नद्यापो नुट् / अट्कुप्वाङ्नुम्व्यवाऽपेपि

<sup>123</sup> याडापः / डेराम् नद्याम्नीभ्यः / हल्-ङ्याभ्यो दीर्घात् सुतिस्यपृक्तं हल्

<sup>124</sup> एङ्-ह्रस्वात् संबुद्धेः

<sup>125</sup> लशक्वतद्धिते / सर्वनाम्नः स्याङ्ङ्रस्वश्च / वृद्धिरेचि

<sup>126</sup> सर्वनाम्नः स्याङ्ङ्रस्वश्च / लशक्वतद्धिते / वृद्धिरेचि

<sup>127</sup> आमि सर्वनाम्नः सुट्

<sup>128</sup> सर्वनाम्नः स्याङ्ङ्रस्वश्च / डेराम् नद्याम्नीभ्यः / वृद्धिरेचि

<sup>129</sup> शेषोऽयसखि



1-3/2-3	:	H + jas/śas → H + as → harayaḥ [6.1.78] <sup>131</sup>
3-1	:	H + ṭā → H + nā → hariṇā [7.3.119/8.4.2] <sup>132</sup>
3-2/4-2/5-2	:	H + bhyām → haribhyām
3-3	:	H + bhis → haribhiḥ
4-1	:	H + ñe → hare + e → harayaye [7.3.111/6.1.78] <sup>133</sup>
4-3/5-3	:	H + bhyas → haribhyaḥ
5-1/6-1	:	H + ñasi/ñas → hare + as → hareḥ [7.3.111/6.1.97] <sup>134</sup>
6-2/7-2	:	H + os → haryoḥ [6.1.74] <sup>135</sup>
6-3	:	H + ām → hari + n + ām → harī + nām → harīṇām [7.1.54/6.4.3/8.4.2] <sup>136</sup>
7-1	:	H + ñi → hara + au → harau [7.3.118/6.1.85] <sup>137</sup>
7-3	:	H + sup → hari + ṣu [8.3.59] <sup>138</sup> → hariṣu
8-1	:	he H + su → he hare + s [7.3.108] <sup>139</sup> → he hareḥ
8-2/8-3	:	forms are like 1-2/1-3.

‘इ’ and ‘उ’ ending feminine:

“The feminine words ending in ‘i’, ‘u’ are optionally termed ‘*nadī*’ if a case affix with an indicative ‘*na*’ follows. Such case affixes are ‘*ñe*’ (4-1), ‘*ñasi*’ (5-1), ‘*ñas*’ (6-1) and ‘*ñi*’ (7-1). The remaining forms will be declined like their masculine counterparts”<sup>140</sup> -

2-3	:	mati + śas → mati + as → matīḥ [1.3.8/6.1.102] <sup>141</sup>
-----	---	---

<sup>130</sup> प्रथमयोः पूर्वसवर्णः

<sup>131</sup> एचोऽयवायावः

<sup>132</sup> अच्च घेः / अट्कुप्वाङ्नुम्व्यवायेपि

<sup>133</sup> घेङिति / एचोऽयवायावः

<sup>134</sup> घेङिति / अतो गुण

<sup>135</sup> एकः पूर्वपरयोः

<sup>136</sup> ह्रस्व नद्यापो नुट् / नामि / अट्कुप्वाङ्नुम्व्यवायेपि

<sup>137</sup> औत् / अन्तादिवच्च

<sup>138</sup> आदेश प्रत्यययोः

<sup>139</sup> ह्रस्वस्य गुणः

<sup>140</sup> Jha, Girish Nath, M.Phil. dissertation “Morphology of Sanskrit Case Affixes A computational Analysis” page-31.

<sup>141</sup> लशक्वतद्धिते / प्रथमयोः पूर्वसवर्णः

- 3-1 : mati + ṭā → mati + ā → matyā [1.3.8/6.1.77]<sup>142</sup>
- 4-1 : mati + ṇe → mati + e → mati + ā + e → mati + ei →  
matyai [1.3.8/7.3.112/6.1.87/6.1.74]<sup>143</sup>
- 5-1 : mati + ṇasi → mati + as → mati + ā + as → mati + es →  
mateḥ [1.3.8/7.3.112/6.1.87/6.1.74]<sup>144</sup>
- 6-1 : mati + ṇas → mati + as → mati + ā + as → mati + es →  
mateḥ [1.3.8/7.3.112/6.1.87/6.1.74]<sup>145</sup>
- 7-1 : mati + ṇi → mati + i → mati + ām → mati + ā + ām → mati + es →  
matyām [1.3.8/7.3.117/7.3.112/6.1.87/6.1.74]<sup>146</sup>

### ‘इ’ and ‘उ’ ending neuter

‘i’, ‘u’ ending feminine ‘vāri’ and ‘sulu’ are representative of this class.

The forms for generic base ‘vāri’(V) follow-

- 1-1/2-1 : V + su/am → vāri [7.1.23]<sup>147</sup>
- 1-2/2-2 : V + au/auṭ → vāri + śī → vāri + ī → vāri + n + ī →  
vāriṇī [7.1.19/1.3.8/7.1.73/8.4.2]<sup>148</sup>
- 1-3/2-3 : V + jas/śas → V + śi → vāri + n + i → vāriṇi [7.1.20/7.1.73/8.4.2]<sup>149</sup>
- 3-1 : V + ṭā → V + nā → vāriṇā [7.3.119/8.4.2]<sup>150</sup>
- 3-2/4-2/5-2 : V + bhyām → vāribhyām
- 3-3 : V + bhis → vāribhiḥ
- 4-1 : V + ṇe → vāri + n + e → vāriṇe [7.1.73/8.4.2]<sup>151</sup>
- 4-3/5-3 : V + bhyas → vāribhyaḥ

<sup>142</sup> लशक्वतद्धिते / इकोयणचि

<sup>143</sup> लशक्वतद्धिते / आप्नद्धाः / आद् गुणः/ एकः पूर्वपरयोः

<sup>144</sup> लशक्वतद्धिते / आप्नद्धाः / आद् गुणः/ एकः पूर्वपरयोः

<sup>145</sup> लशक्वतद्धिते / आप्नद्धाः / आद् गुणः / एकः पूर्वपरयोः

<sup>146</sup> लशक्वतद्धिते / इदुद्भ्याम् / आप्नद्धाः / आद् गुणः / एकः पूर्वपरयोः

<sup>147</sup> स्वमोर्नपुंसकात्

<sup>148</sup> नपुंसकात् / लशक्वतद्धिते / इकोऽचि विभक्तौ / अट्कुप्वाङ्नुम्व्यवायेऽपि

<sup>149</sup> जस् शसोः शिः / इकोऽचि विभक्तौ / अट्कुप्वाङ्नुम्व्यवायेऽपि

<sup>150</sup> अच्च घेः / अट्कुप्वाङ्नुम्व्यवायेऽपि

<sup>151</sup> इकोऽचि विभक्तौ / नामि

5-1/6-1	:	V + nasi → vāri +n + as → vāriṇaḥ [7.1.73/8.4.2] <sup>152</sup>
6-2/7-2	:	V + os → vāri +n + os → vāriṇoḥ [7.1.73/8.4.2] <sup>153</sup>
6-3	:	V + ām → vāri +n + ām → vārī + ām → vārīṇām [7.1.73/6.4.3/8.4.2] <sup>154</sup>
7-1	:	V + ni → vāri +n + i → vārīṇi [7.1.73/8.4.2] <sup>155</sup>
7-3	:	V + sup → vāri +ṣu [8.3.59] <sup>156</sup> → vārīṣu
8-1/8-2/8-3	:	forms are like 1-2/1-3/1-3.

Some notable exceptions to this pattern are ‘sakthi’, ‘asthi’ and ‘dadhi’<sup>157</sup>,

‘ई’, ‘ऊ’ ending masculine:

In this class, some general rules apply and sup are added with base- (1-1) – *h*, (1-2/2-2) – *au*, (1-3) – *aḥ*, (2-1) – *m* [6.1.105], (2-3) – *n*, (3-1) – *e*, (3-2/4-2/5-2) – *bhyām*, (3-3) – *bhiḥ*, (4-1) – *aḥ*, (4-3/5-3) – *bhyaḥ*, (5-1/6-1) – *aḥ*, (6-2/7-2) – *os*, (6-3) – *ām*, (7-1) – *i*, (7-3) – ‘*su*’. Some notable exceptions are – ‘*vāṭpramī*’ is derived by ‘*kvip*’ and not ‘*kit*’ then 2-1, 2-3 and 7-1 will be derived differently<sup>158</sup>.

‘ई’, ‘ऊ’ ending feminine:

‘*ī*’, ‘*ū*’ ending feminine ‘*gaurī*’ and ‘*khalapū*’ are representative of this class.

1-1	:	gaurī + su → gaurī [6.1.66] <sup>159</sup>
1-2/2-2	:	gaurī + au/auḥ → gauryau [6.1.74] <sup>160</sup>
1-3	:	gaurī + jas → gaurī + ā + as → gauryaḥ [7.3.112/6.1.74] <sup>161</sup>
2-1	:	gaurī + am → gaurīm [6.1.103] <sup>162</sup>

<sup>152</sup> इकोऽचि विभक्तौ / नामि

<sup>153</sup> इकोऽचि विभक्तौ / अट्कुप्वाङ्नुम्व्यवायेऽपि

<sup>154</sup> इकोऽचि विभक्तौ / नामि / अट्कुप्वाङ्नुम्व्यवायेऽपि

<sup>155</sup> इकोऽचि विभक्तौ / अट्कुप्वाङ्नुम्व्यवायेऽपि

<sup>156</sup> आदेश प्रत्यययोः

<sup>157</sup> अस्थिदधिसक्थक्षणामनडुदात्तः [7.1.35]

<sup>158</sup> वोपसर्जनस्य [६.४.८२]

<sup>159</sup> हल्-ङ्याभ्यो दीर्घात् सुतिस्स्यपृक्तं हल्

<sup>160</sup> एकः पूर्वपरयोः

<sup>161</sup> आणनद्याः / एकः पूर्वपरयोः

2-3	:	gaurī + śas → gaurīḥ [6.1.102] <sup>163</sup>
3-1	:	gaurī + ṭā → gaurī + ā → gauryā [1.3.8/6.1.74] <sup>164</sup>
3-2/4-2/5-2	:	gaurī + bhyām → gaurībhyām
3-3	:	gaurī + bhis → gaurībhiḥ
4-1	:	gaurī + ñe → gaurī + e → gaurye [1.3.8/6.1.74] <sup>165</sup>
4-3/5-3	:	gaurī + bhyas → gaurībhyaḥ
5-1/6-1	:	gaurī + nasi → gaurī + as → gaurī + ā + as → gauryāḥ [1.3.8/7.3.112/6.1.74] <sup>166</sup>
6-2/7-2	:	gaurī + os → gauryoḥ [6.1.74] <sup>167</sup>
6-3	:	gaurī + ām → gaurī + n + ām → gaurī + ām → gaurīṇām [7.1.73/6.4.3/8.4.2] <sup>168</sup>
7-1	:	gaurī + ñi → gaurī + ā + ñi → gaurī + ā + ām → gauryām [7.3.112/7.3.116/6.1.74] <sup>169</sup>
7-3	:	gaurī + sup → gaurī + ṣu [8.3.59] <sup>170</sup> → gaurīṣu

### ‘ऋ’, ‘ॠ’ and ‘लृ’ ending

The declension of ‘r’ ending stems is comparatively limited being almost entirely composed of derivative nouns formed with the suffix ‘tr’ but it also includes nouns like ‘devr’, ‘nr’, ‘savyaṣṭhr’. The inflection of these stems is quite analogous with those ending in ‘i’ and ‘u’ and its peculiarity consists mainly in the treatment of the stem itself which has a double form, fuller in strong cases and briefer in weak cases<sup>171</sup>

<sup>162</sup> तस्माच्छसो नः पुंसि

<sup>163</sup> प्रथमयोः पूर्वसवर्णः

<sup>164</sup> लशक्वतद्धिते / एकः पूर्वपरयोः

<sup>165</sup> लशक्वतद्धिते / एकः पूर्वपरयोः

<sup>166</sup> लशक्वतद्धिते / आप्नद्धाः / एकः पूर्वपरयोः

<sup>167</sup> एकः पूर्वपरयोः

<sup>168</sup> इकोऽचि विभक्तौ / नामि / अट्कुप्वाङ्नुम्व्यवायेऽपि

<sup>169</sup> आप्नद्धाः / डेराम् नद्याम्नीभ्यः / एकः पूर्वपरयोः

<sup>170</sup> आदेश प्रत्यययोः

<sup>171</sup> W.D. Whitney “Sanskrit Grammar” pp-137

For ‘r’ ending ‘dhātr’ special rules 7.3.110/7.1.94/6.1.11/8.2.24/ will apply for the generating process and remaining forms will be processed by general rules. However, ‘nr’ is different in 8-1 where it is ‘he naḥ’. The word ‘pitṛ’ will be according to ‘dhātr’ and only different in (1-2/2-2) ‘pitṛ’ by the rule 7.3.110.

‘r’ ending feminine ‘svsṛ’, ‘nanādr’, ‘dhuhitr’, ‘yātr’, etc. are declined by general rule and apply rule 6.4.11 and the word ‘mātr’ is like ‘pitṛ’ but only 2-3 is different ‘mātrḥ’ (not mātrn).

Only nominative and accusative ‘r’ ending neuter forms are different from masculine. These forms are declined like ‘dhātr’ and remaining forms will be declined like ‘r’ ending masculine forms.

‘i’ ending forms are derived by general addition of suffixes and *sandhi* rules for example - ‘kr’, ‘krau’, ‘karaḥ’ etc.

The words ‘gamlr’, ‘śakr’ etc. are ‘lr’ ending. These are declined as ‘gama’, ‘śak’ 1-1 by the rule: 7.1.94<sup>172</sup>. The forms are ‘gamlu, gamalaḥ (1-2,1-3) by 7.3.110<sup>173</sup>. For 2-3 ‘gamlrn’ and for 5-1/6-1 forms are ‘gamul’/ ‘śakul’ by rules 6.1.111 / 1.1.51<sup>174</sup>. The remaining forms are by general rules.

### ‘ए’, ‘ओ’, ‘ऐ’ and ‘औ’ ending

‘e’ ending forms are found in masculine only. They are derived by general addition of *sandhi* rules, for example ‘se’, ‘sayau’, ‘sayah’ etc. For ‘ai’ ending masculine and feminine stems the ‘ai’ is substituted by ‘ā’ [7.2.85]<sup>175</sup>. The remaining forms are derived by simple *sandhi* rules.

For neuter ‘e’, ‘ai’, ‘o’, ‘au’ (ec) ending stems, the ‘ik’ would be substituted by rule 1.148<sup>176</sup>. Therefore ‘atirai’, ‘pradyau’ become ‘atiri’, ‘pradyu’ etc. and derived like ‘i’ ending.

<sup>172</sup> ऋदुशनस्-पुरुदंसोऽनेहसां च

<sup>173</sup> ऋतो डि-सर्वनाम्स्थानयोः

<sup>174</sup> उरण् रपरः

<sup>175</sup> रायो हलि

The ‘o’ ending stems will go through ‘*vrddhi*’[7.1.90/7.2.155]<sup>177</sup> for strong case endings. Therefore ‘*gauḥ*’, ‘*gāvau*’, ‘*gāvaḥ*’, ‘*he gauḥ*’, (1-1, 1-2, 1-3, 8-1) and ‘*gām*’, ‘*gāḥ*’ etc. (2-1, 2-3 ) is derived by rule 6.1.91<sup>178</sup> and remaining forms are derived by ordinary *sandhi* rules.<sup>179</sup>

## 2.2.2 Consonant ending

Consonant ending bases show related patterns of derivation as far as the forms of the case endings are concerned. Some forms are generated by some changes in the bases while other changes take place in the suffixes.

The ‘h’ ending masculine forms are derived by some changes in the base in nominative singular (1-1), instrumental plural (3-3) and instrumental, dative and ablative duals (3-2/4-2/5-2). We take ‘*lih*’ which is representative of this class-

1-1	:	lih + su → lih → liḥ/liḍ [6.1.68/8.2.31/8.2.39/8.4.55] <sup>180</sup>
1-2/2-2	:	lih + au/auḥ → lihu
1-3/2-3/5-1/6-1	:	lih + jas/śas/ñasi/ñas → lih + as → lihaḥ [1.3.8] <sup>181</sup>
2-1	:	lih + am → liham
3-1	:	lih + tā → lih + ā → lihā
3-2/4-2/5-2	:	lih + bhyām → liḥbhyām/liḍbhyām [8.2.31/8.2.39/8.4.55] <sup>182</sup>
3-3	:	lih + bhis → liḥbhiḥ/liḍbhiḥ [8.2.31/8.2.39/8.4.55] <sup>183</sup>
4-1	:	lih + ne → lihe
4-3/5-3	:	lih + bhyas → liḥbhyaḥ/liḍbhyaḥ [8.2.31/8.2.39/8.4.55] <sup>184</sup>
6-2/7-2	:	lih + os → lihoḥ

<sup>176</sup> एच् इग्नस्वादेशे

<sup>177</sup> गोतो णित् / षट्चतुर्भ्यश्च

<sup>178</sup> औतोऽप्शसोः

<sup>179</sup> Jha, Girish Nath, M.Phil. dissertation “Morphology of Sanskrit Case Affixes A computational Analysis” page-36.

<sup>180</sup> हल्-इयाभ्यो दीर्घात् सुतिस्त्वपृक्तं हल् / होढः / स्कोः संयोगाद्योरन्ते च / खरि च

<sup>181</sup> लशक्वतद्धिते

<sup>182</sup> होढः / झलां जशोऽन्ते / खरि च

<sup>183</sup> होढः / झलां जशोऽन्ते / खरि च

<sup>184</sup> होढः / झलां जशोऽन्ते / खरि च

6-3	:	lih + ām → lihām
7-1	:	lih + ñi → lihi
7-3	:	lih + sup → liṭsu/liḍsu [8.2.31/8.2.39/8.4.55] <sup>185</sup>
8-1	:	he lih + su → he liṭ/ he liḍ [8.2.31/8.2.39/8.4.55] <sup>186</sup>
8-2/8-3	:	like 1-2 and 1-3.

Some exceptions are there in following cases (1-1/3-3/3-2/4-2/5-2/4-3/5-3). The last character of ‘*duh*’ changes to ‘*k*’/‘*g*’ and remaining forms are like ‘*lih*’. Word ‘*viśvāh*’ changes to ‘*ṭ*’/‘*ḍ*’ ending and applies ‘*samprasāraṇa*’<sup>187</sup> rule for generating other forms.

The ‘*h*’ ending feminine forms are generated like ‘*h*’ ending masculine but nominative singular ‘*h*’ is changed to ‘*t*’ or ‘*d*’- *upānat/upānad* (1-1), instrumental plural- *upānadbhiḥ/upānatbhiḥ* (3-3), instrumental, dative and ablative duals- *upānadbhyām/upānatbhyām* (3-2/4-2/5-2) and dative and ablative plural- *upānadbhyaḥ/upānatbhyaḥ* (4-3/5-3) by the rule 8.2.34<sup>188</sup>. Some exceptions are also notable - ‘*h*’ ending word ‘*uṣṇih*’ changes to ‘*k*’/‘*g*’ in (1-1/3-3/3-2/4-2/5-2/4-3/5-3).

The ‘*h*’ ending neuter bases ‘*svanḍuh*’ are derived like masculine but there are some differences in nominative and accusative cases - ‘*h*’ changes to ‘*ṭ*’/‘*ḍ*’ in 1-1 and 2-1, for the nominative and accusative, the dual suffix is ‘*au*’/‘*auṭ*’ which changes to ‘*śt*’ by rule 7.1.19<sup>189</sup>. For nominative and accusative plural, suffix ‘*jas*’/‘*śas*’ changes to ‘*śi*’ and gets two augment ‘*ām*’ and ‘*num*’ and applies ‘*yaṇa sandhi*’<sup>190</sup> rule and vowel *sandhi* rule for generating ‘*svanaḍvāmhi*’. Four rules apply in this process [7.1.20/41/7.1.98/7.1.72/6.1.77/8.3.24]<sup>191</sup>

For ‘*v*’ ending masculine ‘*sudiv*’ 1-1, the suffix is ‘*su*’ which change to ‘*aut*’ by rule 7.1.84<sup>192</sup> and applies *sandhi* rule for generating ‘*sudyau*’ and in the 3-2/4-2/5-2/3-3/4-3/5-3 cases, the last vowel of ‘*sudiv*’ is converted to ‘*u*’. The applies the ‘*yaṇa sandhi*’<sup>193</sup> rule to

<sup>185</sup> होढः / झलां जशोऽन्ते / खरि च

<sup>186</sup> होढः / झलां जशोऽन्ते / खरि च

<sup>187</sup> सम्प्रसारणाच्च

<sup>188</sup> नहो धः

<sup>189</sup> नपुंसकाच्च

<sup>190</sup> इको यणचि

<sup>191</sup> जस् शसोः शिः / चतुरनडुहोरामुदात्तः / नपुंसकस्य झलचः / इको यणचि / नश्चापदान्तस्य झलि

<sup>192</sup> दिव औत्

<sup>193</sup> इको यणचि

generate ‘*sudyubhyām/ sudyubhiḥ/sudyubhyaḥ*’ etc. All ‘*v*’ ending feminine and neuter forms are declined like masculine.

The last character of ‘*m*’ ending masculine base changes to ‘*n*’ by the rule 8.2.64<sup>194</sup> in (1-1/3-3/3-2/4-2/5-2/4-3/5-3) and generates ‘*praśān*’/ *praśānbhyām / praśānbhiḥ / praśānbhyaḥ*’ etc. All remaining forms are like ‘*lih*’. ‘*m*’ ending masculine and feminine pronoun ‘*kim*’ declines like ‘*sarva*’, and ‘*sarvā*’. ‘*ka*’ replaces ‘*kim*’ in all the cases and generates ‘*kaḥ/kā, kau/ke, ke/kāḥ*’ etc. like ‘*sarva/sarvā*’. Some exceptions are there which are derived by special rules like ‘*idam*’. ‘*m*’ ending neuter pronoun ‘*kim*’ declines like ‘*sarva*’. Only ‘*ka*’ replaces ‘*kim*’ in some cases except (1-1/2-1) and generates ‘*kim, ke, kāni*’ etc. like ‘*sarva*’.

For ‘*n*’ / ‘*ṇ*’ ending stems ‘*rājan*’, the penultimate vowel will get length by 6.4.8<sup>195</sup>, and ‘*n*’ and ‘*su*’ will be deleted by 8.2.7 and 6.1.68<sup>196</sup>. As a result we have *rājā* (1-1) from ‘*rājan*’. The penultimate ‘*ā*’ is deleted in 2-3/5-1/6-1 by the rule 6.4.134<sup>197</sup>, and forms like ‘*rājñah*’, *rājñā* (3-1), *rājñe* (4-1), *rājñoh* (6-2/7-2), *rājñām* (6-3) and *rājñi* (7-1) are obtained. For ‘*bhyām*’, ‘*n*’ will be deleted by 8.2.2<sup>198</sup> (except in (7-1) and (8-1)). For ‘*yajvan*’ and ‘*brāhman*’, the ‘*a*’ of ‘*an*’ is not deleted in 2-3, 3-1, 4-1, 5-1, 6-1, 6-2, 7-1, 7-2 by the rule 6.4.137<sup>199</sup>. For ‘*vṛtrahan*’, 2-3 and 3-1 will be ‘*vṛtraghnah*’ and ‘*vṛtraghnā*’ by the rule 6.4.134/7.3.54/8.4.22<sup>200</sup>. In the same way, ‘*śārṅgin*’, ‘*yaśasvin*’ etc. can be obtained. The process of ‘*maghvan*’ is done by rule 6.4.128/7.1.70/6.4.14/6.1.68/6.4.133/6.1.108/6.1.87<sup>201</sup>. For the ‘*divan*’ or that ending in ‘*van*’, will be like ‘*rājan*’, but accusative plural (2-3) will be ‘*divanā*’. For ‘*śvan*’ the forms are ‘*śunah*’ (2-3), ‘*śunā*’ (3-1), ‘*śvabhyām*’ etc. ‘*yuvan*’ will be derived by 6.1.37<sup>202</sup>. ‘*a*’ is substituted for the final of ‘*pathin*’, ‘*mathin*’ and ‘*rbhukṣin*’ before ‘*su*’ by 7.1.95<sup>203</sup> and ‘*a*’ comes for ‘*i*’ by 7.1.86<sup>204</sup> in strong cases<sup>205</sup>.

<sup>194</sup> मो नो धातोः

<sup>195</sup> सर्वनाम् स्थाने चासम्बुद्धौ

<sup>196</sup> न लोपः प्रातिपदिकान्तस्य / हल्-ङ्याभ्यो दीर्घात् सुतिस्यपृक्तं हल्

<sup>197</sup> अल्लोपोनः

<sup>198</sup> न लोपः सुप्-स्वर्-संज्ञा-तुग्विधिषु कृति

<sup>199</sup> न संयोगाद्वन्तात्

<sup>200</sup> अल्लोपोनः / हो हन्तेर्जिन्नेषु / हन्तेरत्पूर्वस्य

<sup>201</sup> मघवा बहुलम् / उगिदचां सर्वनामस्थानेऽधातोः / अत्वसन्तस्य चाधातोः / हल्-ङ्याभ्यो दीर्घात् सुतिस्यपृक्तं हल् / श्व-युव-मघोनामतद्धिते / सम्प्रसारणाच्च / आद गुणः

<sup>202</sup> न सम्प्रसारणे सम्प्रसारणम्

<sup>203</sup> तृज्वत् क्रोष्टुः

<sup>204</sup> इतोऽत् सर्वनामस्थाने

<sup>205</sup> Jha, Girish Nath, M.Phil. dissertation “Morphology of Sanskrit Case Affixes A computational Analysis”



For ‘r’ ending base ‘*catur*’, the forms are found only in plural because this is a plural numeral. Derivation process is as follows -

- 1-3/2-3 : *catur* + *jas* → *catu* + *a* + *r* + *jas* → *catur* + *aḥ* → *caturah*  
[7.1.98/1.3.7]<sup>206</sup>
- 3-3 : *catur* + *bhis* → *caturbhiḥ*
- 4-3/5-3 : *catur* + *bhyas* → *caturbhyaḥ*
- 6-3 : *catur* + *ām* → *catur* + *nām* →  
*caturṇām/caturṇām* [7.1.55/8.4.1/8.4.45]<sup>207</sup>
- 7-3 : *catur* + *sup* → *catursu*

‘*dh*’ ending stems are derived by general rules. ‘*ś*’ (palatal) and ‘*ṣ*’ (cerebral) ending bases are derived by following changes:

*ś* → *k* [8.2.62] → *ṣ* [8.2.36] → *ḍ* [7.2.39] → *k* [8.4.56]<sup>208</sup>

Thus for ‘*viś*’ 1-1 will be ‘*viṭ/vid*’. For ‘*naś*’ 1-1 will be ‘*nak/nag/naṭ/naḍ*’ (four optional forms).

‘*ṣ*’ ending stems (like ‘*pipaṭhiṣ*’) will undergo following changes in 1-1-

*pipaṭhiṣ* + *su* → *pipaṭhir* → *pipaṭhiḥ* [6.1.68/8.2.66/8.2.76]<sup>209</sup>

For ‘*adas*’, the following processing will take place:

- 1-1 : *adas* + *su* → *ad* + *au* → *as* + *au* → *asu* [7.2.107/7.2.106]<sup>210</sup>
- 1-2 : *adas* + *au* → *ad* + *au* → *adau* → *amū* [7.2.102/6.1.96/6.1.102]<sup>211</sup>
- 1-3 : *adas* + *jas* → *ad* + *śī* → *ad* + *ī* → *amī* [7.1.17/6.1.87/8.2.81]<sup>212</sup>
- 2-1 : *adas* + *am* → *adam* → *amum* [8.2.80]<sup>213</sup>

<sup>206</sup> चतुरनडुहोरामुदात्तः / चुटू

<sup>207</sup> षट्चतुर्भ्यश्च / रषाभ्याम् नो णः समानपदे / अचो रहाभ्याम् द्वे

<sup>208</sup> क्विप्प्रत्ययस्य कुः / व्रश्च-भ्रश्च-सृज-मृज-यज-राज-भ्राज-च्छ-शां षः / ली-लोर्नुग्लुकावन्यतरस्यां स्नेहविपातने / वाऽवसाने

<sup>209</sup> हल्-ङ्याभ्यो दीर्घात् सुतिस्त्र्यपुक्तं हल् / ससजुषो रुः / वोरूपधाया दीर्घ इकः

<sup>210</sup> अदस औ सुलोपश्च / तदोः सः सावनन्त्ययोः

<sup>211</sup> त्वदादीनाम् / उस्त्र्यपदान्तात् / प्रथमयोः पूर्वसवर्णयोः

<sup>212</sup> जसः शी / आद् गुणः / एत ईद् बहुवचने

<sup>213</sup> अदसोऽसेर्दादु दो मः

When ‘*adas*’ is changed to ‘*amu*’ it gets the ‘*ghi*<sup>214</sup>’ designation. Thus, in 3-1 ‘*amunā*’ 3-2/4-2/5-2 ‘*amībhyām*’, 3-3 ‘*amībhiḥ*’, 4-3/5-3 ‘*amībhyaḥ*’, 6-2/7-2 ‘*amuyoh*’, 6-3 ‘*amiṣām*’. ‘*d*’ ending words ‘*asmad*’ and ‘*yusmad*’ are similar and derived by following process:

- 1-2 : asmad + su → asmad + am → ah + ad + am → ahad + am → aham  
[7.1.28/7.2.94/6.1.97/7.2.90]<sup>215</sup>
- 1-2/2-2 : asmad + au/auṭ → asmad + am → āva + ad + am → āvad + am → āva a + ā + am → āvām [7.1.28/7.2.92/7.2.88/6.1.101]<sup>216</sup>
- 1-3 : asmad + jas → asmad + am → vaya + ad + am → vayad + am → vayam  
[7.1.28/7.2.93/6.1.97/7.2.90]<sup>217</sup>
- 2-1 : asmad + am → ma + ad + am → mad + am → ma + ā + am → mām  
[7.2.97/6.1.97/7.2.87/6.1.101]<sup>218</sup>
- 2-3 : asmad + śas → asmad + am → asmad + nas → asma + ā + am → asmānas → asmān [7.2.97/7.1.29/7.2.87/6.1.101]<sup>219</sup>
- 3-1 : asmad + ṭā → ma + ad + ā → mad + ā → may + ā → mayā  
[7.2.97/6.1.97/7.2.89]<sup>220</sup>
- 3-2/4-2/5-2 : asmad + bhyām → āva + ad + bhyām → āvad + bhyām → āvābhyām  
[7.2.92/6.1.97/7.2.86]<sup>221</sup>
- 3-3 : asmad + bhis → asmā + bhis → asmābhiḥ [7.2.86/6.1.101]<sup>222</sup>
- 4-1 : asmad + ñe → asmad + am → mahyad + am → mahyam  
[7.1.28/7.2.95/7.2.87/7.2.90]<sup>223</sup>
- 4-3 : asmad + bhyas → asmabhyam [7.1/30/7.2.87/6.1.97]<sup>224</sup>
- 5-1 : asmad + nasi → asmad + at → ma ad + at → asmat [7.1.32/7.2.93/7.2.90]<sup>225</sup>
- 5-3 : asmad + bhyas → asmad + at → asmat [7.1.31/7.2.90]<sup>226</sup>

<sup>214</sup> शेषोध्यसखि

<sup>215</sup> डे प्रथमयोरम् / त्वाहौ सौ / अतो गुणे / शेषे लोपः

<sup>216</sup> डे प्रथमयोरम् / युवावौ द्विवचने / प्रथमायाश्च द्विवचने भाषायाम् / अकः सवर्णे दीर्घः

<sup>217</sup> डे प्रथमयोरम् / युव वयौ जसि / अतो गुणे / शेषे लोपः

<sup>218</sup> त्व मावेकवचने / अतो गुणे / द्वितीयायां च / अकः सवर्णे दीर्घः

<sup>219</sup> त्व मावेकवचने / शसो न / द्वितीयायां च / अकः सवर्णे दीर्घः

<sup>220</sup> त्व मावेकवचने / अतो गुणे / योऽचि

<sup>221</sup> युवावौ द्विवचने / अतो गुणे / युस्मदस्मदोरनादेशे

<sup>222</sup> युस्मदस्मदोरनादेशे / अकः सवर्णे दीर्घः

<sup>223</sup> डे प्रथमयोरम् / तुभ्य-महौ डयि / द्वितीयायां च / शेषे लोपः

<sup>224</sup> भ्यसो भ्यम् / द्वितीयायां च / अतो गुणे

<sup>225</sup> एकवचनस्य च / युव वयौ जसि / शेषे लोपः

- 6-1 : asmad + ñas → mama + ad + ñas → mamad + a → mama  
[7.2.96/7.1.27/7.2.90]<sup>227</sup>
- 6-2/7-3 : asmad + os → aava + ad + os → aavad + os → aavayoH  
[7.2.92/6.1.97/7.2.89]<sup>228</sup>
- 6-3 : asmad + aam → asmad + ākam → asmākam [7.1.33/7.2.90]<sup>229</sup>
- 7-1 : asmad + ni → ma + ad + i → mad + i → mayi [7.2.97/6.1.97/7.2.89]<sup>230</sup>
- 7-3 : asmad + sup → asmāsu [7.2.86]<sup>231</sup>

## 2.3. Subanta recognition mechanism

First of all, the system checks for punctuations. Then the *avyayas* and the verbs are recognized. After the recognition of these *padas*, the system recognizes all remaining words as *subanta* and sends for the analysis process.

### 2.3.1 Recognition of punctuations and non-subanta words

System will recognize punctuations and tag them with \_PUNCT. If the input has any extraneous characters, then the input word will be cleaned from these elements (normalized) so that only Devanāgarī Sanskrit input text is sent to the analyzer. For example - रा/&^%#@#म्:, बा,"":-लकः → रामः, बालकः .

### 2.3.2 avyaya recognition

Sanskrit sentence must have a *tinanta-pada* and can have one or more *subanta-padas* (including *avyayas*). We have stored around 524 *avyayas* in the following format-

Avyaya_id	avyaya
1	अ
2	अकस्मात्

<sup>226</sup> पञ्चम्या अत् / शेषे लोपः

<sup>227</sup> तव-ममौ डसि / युस्मदस्मद्भ्याम् डसोऽश् / शेषे लोपः

<sup>228</sup> युवावौ द्विवचने / अतोऽगुणे / योऽचि

<sup>229</sup> साम् आकम् / शेषे लोपः

<sup>230</sup> त्व मावेकवचने / अतोऽगुणे / योऽचि

<sup>231</sup> युस्मदस्मदोरनादेशे

3	अकाण्डे
4	अग्निसात
5	अग्नी
6	अघोः
7	अङ्ग
8	अजस्रम
9	अञ्जसा
10	अतः
11	अृत
12	अतीव
13	अत्र
14	अथ
15	अथकिम

**Table-2.1: Avyaya recognition**

If an input word is found in the avyaya database, it is labeled ‘\_AV’, and not sent to the subanta analyzer for further processing.

### 2.3.3 Verb (*tinanta*) recognition

Sanskrit verb forms are very complex they carry tense, aspect, number information all in the inflection forms. Sanskrit has about 2000 verb roots classified in 10 morphological and semantic classes. Further, these can have *ātmanepadī* and *parasmaipadī* forms in 10 *lakāra* and 3 x 3 persons and numbers combinations and can also be potentially. There are 12 secondary suffixes, which are added to verb roots and create new verb roots. Mishra & Jha (2004) have done a rough calculation of all potential verb forms in Sanskrit to be more than 10,29,60,000. Since storing all these verb forms would have been difficult, we have stored 500 commonly used verb roots and their forms. Thus we have around 90,000 verb forms stored in unicode. A sample listing follows –

dhātu_id	dhātu	lat_pra_eka	lat_pra_dvi	lat_pra_bahu
----------	-------	-------------	-------------	--------------

1	ठू	भवति	भवतः	भवन्ति
2	एध्	एधते	एधेते	एधन्ते
3	स्पर्ध्	स्पर्धते	स्पर्धेते	स्पर्धन्ते
4	गाध्	गाधते	गाधेते	गाधन्ते
5	बाध्	बाधते	बाधेते	बाधन्ते
6	नाध्	नाथते	नाथेते	नाथन्ते
7	नाध्	नाधते	नाधेते	नाधन्ते
8	दध्	दधते	दधेते	दधन्ते
9	स्कुदि	स्कुन्दते	स्कुन्देते	स्कुन्दन्ते
10	वदि	वन्दते	वन्देते	वन्दन्ते

**Table-1: verb forms**

If an input is found in the verb database, it is labeled ‘\_VERB’ and not sent to the analyzer for further processing

#### 2.3.4 *Subanta* recognition

Thus the *subanta-padas* in Sanskrit text are identified by a process of exclusion. After the verbs and avyayas are identified by their lexical pattern matching search, the remaining *padas* in the text are labeled *subanta-padas*.

## ***Chapter-III***

### ***SUBANTA ANALYSIS***

### 3.1 Vowel ending *prātipadika*

Vowel ending *prātipadikas* follow the general pattern mostly. However, there is some variation if we move across gender and categories of the *prātipadikas* as seen below -

#### 3.1.1 Nominative singular/dual/plural

System will recognize the *vibhakti* marker as the end character of *padas*. ‘:’ is found in nominative singular (1-1) for example, रामः, श्यामः, सर्वः, भरतः एकः. The system will isolate ‘:’ and search for analysis in the *sup* database.

In case of nominative and accusative dual (1-2/2-2), *prātipadika* forms will be ‘ः’ ending. System will recognize ‘ः’. For example, रामौ, श्यामौ, सर्वौ, एकौ. The system will isolate ‘ः’ and search for analysis by matching in the *sup* database.

In case of nominative plural (1-3), the *prātipadika* forms will be ‘ः’ ending. For example, रामाः, श्यामाः, भरताः एकाः. The system will isolate ‘ः’ and search for analysis by matching in the *sup* database.

All vowel ending nominative singular/dual/plural patterns have been displayed in the following table -

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
1-1	अ, इ, ई, उ, ऊ, औ	:		रामः, हरिः, पपीः, भानुः, खलपूः, ग्लौः	neuter gender अ ending
1-1	आ, ऋ, ए	ः	ः	रमा, लता, पितृ	विश्वपा
1-1	ऐ	ः	ः	राः	
1-2	अ	ः		रामौ	

1-2	आ	ते	।	रमा, विश्वपा	
1-2	इ	नी	ि	हरी, मती	
1-2	ई	यौ	ी	नद्यौ	
1-2	उ	ू	ु	भानू	
1-2	ऊ	वौ	ू	बधू	
1-2	ऋ	ारौ	ृ	पितरौ, धातारौ	
1-2	ओ	ावौ	े	गावौ	
1-2	ऐ	ायौ	ै	रायौ	
1-2	औ	ावौ	ै	ग्लावौ	
1-3	अ, आ,	।:	।	रामा:,रमा:	सर्वादि गण-सर्वे
1-3	इ	ाय:	ि	हरय:	
1-3	ई	ाय:/:	ी	पय्य:/नदी:	
1-3	उ	ाव:	ु	गुरव:	
1-3	ऊ	व:	ू	गाव:	
1-3	ऋ	ार:	ृ	धातार:	
1-3	ओ	ाव:	े	गाव:	
1-3	ऐ	ाय:	ै	राय:	
1-3	औ	ाव:	ै	ग्लाव:	

**Table 3.1 : Nominative analysis pattern**

### 3.1.1.1 Special cases/exceptions

Some exceptions are found in nominative plural pronouns for example, *sarva*. System analyzes this type of *subanta* forms with the help of example, base. In this database, many *subanta* forms are stored with analysis, which can not be handled through general rules.



Subanta	Analysis
सर्वे	सर्व + जस् प्रथमा बहुवचन

**Table 3.2 : Nominative vowel ending exception analysis pattern**

### 3.1.2 Accusative singular/dual/plural

In case of accusative singular (2-1), *prātipadika* forms will be ‘म्’ ending. For example, रामम्, श्यामम्, भरतम् एकम् In this case, it will check the last character search for analysis by matching in the sup database.

In case of accusative case plural (2-3), *prātipadika* forms will be ‘न्’ ending. For example, रामान्, श्यामान्, भरतान् In this case, it will check the last character and search for analysis by matching in the sup database.

All vowel ending accusative singular/dual/plural patterns have been displayed in following table-

Kāraka - Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
2-1	अ, आ, इ, ई, उ, ऊ	म्		रामम्, हरिम्, नदीम्, भानुम्, खलपूम्	
2-1	ऋ, ओ	त्	त्	धातारम्	
2-1	ऐ	यम्	यै	रायम्	
2-1	औ	वम्	वै	ग्लावम्	
2-2	अ, आ, इ, ई, उ, ऊ, ऋ, ए, ऐ, ओ, औ	न्			

2-2	आ	त	ते	रमे	
2-2	इ	नी	नि	हरी	
2-2	ई	यौ	यी	पपी	
2-2	उ	तु	तु	गुरू	
2-2	ऊ	वौ	वु	हूहू	
2-2	ऋ	तरौ	तु	धातारौ	
2-2	ओ	तवौ	ते	गावौ	
2-2	ऐ	तयौ	तु	रायौ	
2-2	औ	तवौ	तै	ग्लावौ	
2-3	अ	तन्		रामान्	
2-3	आ	तः	ता	रमाः	
2-3	इ	निन्	नि	हरीन्	
2-3	ई	निन्	नीन्	पपीन्	नदी
2-3	उ	तून्	तु	गुरून्	
2-3	ऊ	तून्	तून्	हूहून्	
2-3	ऋ	तृन्	तृन्	धातृन्	
2-3	ओ	तः	ते	गाः	
2-3	ऐ	तयः	तु	रायः	
2-3	औ	तवः	तै	ग्लावः	

**Table-3.3 : Accusative vowel ending analysis pattern**

### 3.1.2.1 Special cases/exceptions

For 'ई' ending feminine bases, we have created special paradigms in this format -

2-3	ई	ीः	ी	नदीः	
2-3	ई	ीः	ी	गौरीः	

**Table-3.4 : Accusative exceptional analysis pattern**

### 3.1.3 Instrumental singular/ plural and instrumental dative & ablative dual

In case of instrumental singular (3-1), *prātipadika* forms will be 'ेन / ेण' ending. For example, रामेण, श्यामेन, भरतेन एकेन. In this case, it starts checking from last character of the *subanta* forms and search for analysis by matching in the sup database.

In case of instrumental, dative and ablative dual (3-2/4-2/5-2), the *prātipadika* forms will be 'त्भ्याम्' ending. System will isolate 'त्भ्याम्' and search for analysis by matching in the sup database.

In case of instrumental plural (3-3), the *prātipadika* forms will be 'ैः' ending. For example, रामैः, श्यामैः, तैः, भरतैः. The system will isolate 'ैः' and search for analysis by matching in the sup database.

All vowel ending instrumental singular/dual/plural patterns have been displayed in the following table-

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
3-1	अ	ेन / ेण		रामेण	
3-1	आ	या		रमया	
3-1	इ	णा/ना		हरिणा	
3-1	ई	या	ी	पय्या	
3-1	उ	णा/ना		गुरुणा	
3-1	ऊ	वा	ँ	हूँवा	

3-1	ऋ	त्रा/रा	ॠ	धात्रा	
3-1	ओ	त्वा	ॡ	गवा	
3-1	ऐ	त्या	ॢ	राया	
3-1	औ	त्वा	ॣ	ग्लावा	
3-2	अ, ऐ	भ्याम्		रामाभ्याम्, राभ्याम्	
3-2	आ, इ, ई, उ, ऊ, ऋ, ओ, औ	भ्याम्			
3-3	अ	भैः	।	रामैः	
3-3	आ, इ, ई, उ, ऊ, ऋ, ओ, औ	भिः		रमाभिः, नदीभिः, गुरुभिः, पितृभिः	
3-3	ऐ	भिः	॥	राभिः	

**Table-3.5 : Instrumental vowel ending analysis pattern**

### 3.1.4 Dative singular and dative & ablative plural

In dative singular (4-1), *prātipadika* forms will be ‘त्य’ ending. For example, रामाय, श्यामाय, भरताय. In this case, it will isolate ‘त्य’ and search for analysis by matching in the sup database.

In dative and ablative plural (4-3/5-3), the *prātipadika* forms will be ‘भ्यः’ ending. For example, रामेभ्यः, श्यामेभ्यः, तेभ्यः, भरतेभ्यः. In this case, isolate ‘भ्यः’ and search for analysis by matching in the sup database.

All vowel ending dative singular/dual/plural patterns have been displayed in the following table -

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
4-1	अ	ाय		रामाय	सर्वादि गण- सर्वस्मै
4-1	आ	यै		रमायै	
4-1	इ	ये	ि	हरये	
4-1	ई	ये	ी	पय्या	
4-1	उ	वे	ु	गुरवे	
4-1	ऊ	वे	ू	हूहे	
4-1	ऋ	त्रे/रे	ृ	धात्रे	
4-1	ओ	वे	े	गवे	
4-1	ऐ	ये	ै	राये	
4-1	औ	वे	ौ	ग्लावे	
4-2	अ, ऐ	भ्याम्		रामाभ्याम्, राभ्याम्	
4-2	आ, इ, ई, उ, ऊ, ऋ, ओ, औ	भ्याम्			
4-3	अ	भ्यः	त्	रामेभ्यः	
4-3	आ, इ, ई, उ, ऊ, ऋ, ओ, औ	भ्यः		रमाभ्यः, हरिभ्यः, नदीभ्यः	

**Table-3.6 : Dative analysis pattern**

### 3.1.4.1 Special cases/exceptions

Some exceptions are found in dative singular pronouns, for example, *sarva*. System analyzes this type of forms with the help of example, base as shown below -

Subanta	Analysis
सर्वस्मै	सर्व + डे चतुर्थी एप्वचन

**Table-3.7 Dat exceptional vowel ending analysis pattern**

### 3.1.5 Ablative singular/dual/plural

In ablative singular (5-1), the *prātipadika* forms will be ‘त् / ट्’ ending. For example, रामात्, श्यामाद्, भस्तात्. In this case, it will isolate ‘त् / ट्’ and search for analysis by matching in the sup database.

All vowel ending ablative singular/dual/plural patterns have been displayed in the following table-

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
5-1	अ	त्/ट्		रामात् / रामाद्	सर्वादि गण- सर्वस्मै
5-1	आ	याः		रमायाः	
5-1	इ	ः	ि	हरेः	
5-1	ई	यः	ी	पय्यः	
5-1	उ	ः	ु	गुरोः	

5-1	ऊ	वः	॒	हूढः	
5-1	ऋ	ः	॒	धातुः	
5-1	॒	॒ः	॒	गोः	
5-1	॒	॒यः	॒	रायः	
5-1	॒	॒वः	॒	ग्लावः	
5-2	अ, ॒	॒भ्याम्		रमाभ्याम्, राभ्याम्	
5-2	आ, इ, ई, उ, ऊ, ऋ, ॒, ॒	भ्याम्			
5-3	अ	॒भ्यः	॒	रमेभ्यः	
5-3	आ, इ, ई, उ, ऊ, ऋ, ॒, ॒	भ्यः		रमाभ्यः, हरिभ्यः, नदीभ्यः	

**Table 3.8 : Ablative vowel ending analysis pattern**

### 3.1.5.1 Special cases/exceptions

Some exceptions are found in ablative singular pronouns, for example,, *sarva*. System analyzes such forms with the help of example, base as shown below -

Subanta	Analysis
सर्वस्मात्	सर्व + डसि पञ्चमी एकवचन

**Table 3.9 : Ablative exception analysis pattern**

### 3.1.6 Genitive singular/dual/plural

In case of genitive singular (6-1), *prātipadika* forms will be ‘स्य’ ending. For example, रामस्य, श्यामस्य, तस्य, भरतस्य .In this case, it will isolate ‘स्य’ and search for analysis by matching in the sup database.

In case of genitive and locative dual (6-2/7-2), the *prātipadika* form will be ‘योः’ ending. For example, रामयोः, श्यामयोः, तयोः, भरतयोः. In this case, it will isolate ‘योः’ and search for analysis by matching in the sup database.

In case of genitive plural (6-3), the *prātipadika* forms will be ‘नाम् / णाम्’ ending. For example, रामाणाम्, श्यामाणाम्, भरतानाम्. In this case, it will isolate ‘नाम् / णाम्’ and search for analysis by matching in the sup database.

All vowel ending genitive singular/dual/plural patterns have been displayed in the following table-

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
6-1	अ	स्य		रामस्य	
6-1	आ	याः		रमायाः	
6-1	इ	इः	ि	हरेः	
6-1	ई	यः	ी	पय्यः	
6-1	उ	तेः	ु	गुरोः	
6-1	ऊ	वः	ू	हूहः	
6-1	ऋ	ः	ृ	धातुः	
6-1	ओ	तेः	े	गोः	
6-1	ऐ	यः	ै	रायः	
6-1	औ	वः	ौ	ग्लावः	
6-2	अ, आ, इ,	योः		रामयोः,	



	ई			रमयोः	
6-2	उ	वोः	उ	गुर्वोः	
6-2	ऊ	वोः	ऊ	हूह्वोः	
6-2	ऋ	त्रोः/रोः	ऋ	धात्रोः	
6-2	ओ	त्वोः	ओ	गवोः	
6-2	ऐ	त्योः	ऐ	रायोः	
6-2	औ	त्वोः	औ	ग्लावोः	
6-3	अ	णाम्/णाम्		रामाणाम्	
6-3	आ	णाम्/नाम		रमाणाम्	
6-3	इ	णीणाम्/णिनाम्	इ	हरीणाम्	
6-3	ई	याम्	ई	पय्याम्	
6-3	उ	तूणाम्/तूनाम्	उ	गुरुणाम्	
6-3	ऊ	वाम्	ऊ	हूह्वाम्	
6-3	ऋ	तृणाम्/तृनाम्	ऋ	धातृणाम्	
6-3	ओ	त्वाम्	ओ	आवाम्	
6-3	ऐ	त्याम्	ऐ	रायाम्	
6-3	औ	त्वाम्	औ	ग्लावाम्	

**Table-3.10 : Genetive vowel ending analysis pattern**

### 3.1.7 Locative singular/dual/plural

In case of locative singular (7-1), *prātipadika* forms will be <sup>३</sup> ending. For example, रामे, श्यामे, भस्ते. In this case it will isolate <sup>३</sup> and search for analysis by matching in the sup database.

In case of locative plural (7-3), *prātipadika* forms will be 'षु' ending. For example, रामेषु, श्यामेषु, भरतेषु. In this case, it will isolate 'षु' and search for analysis by matching in the sup database.

All vowel ending locative singular/dual/plural pattern have been displayed in the following table

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
7-1	अ	॑	॒	रामे	सर्वादि गण- सर्वस्मिन्
7-1	आ	याम्		रमायाम्	
7-1	इ	॑	॒	हरौ	
7-1	ई	याम्	॑	पय्याम्	
7-1	उ	॑	॒	गुरौ	
7-1	ऊ	वाम्	॑	हूह्वाम्	
7-1	ऋ	॒रि	॑	धातरि	
7-1	ओ	॒वि	॑	गवि	
7-1	ऐ	॒यि	॑	रायि	
7-1	औ	॒वि	॑	ग्लावि	
7-2	अ, आ, इ, ई	योः		रामयोः, रमयोः	
7-2	उ	वोः	॒	गुर्वोः	
7-2	ऊ	वोः	॑	हूह्वोः	
7-2	ऋ	त्रोः/रोः	॑	धात्रोः	
7-2	ओ	॒वोः	॑	गवोः	
7-2	ऐ	॒योः	॑	रायोः	

7-2	औ	त्वोः	तै	ग्लावोः	
7-3	अ, इ, ई, उ, ऋ, ए, ओ, ऐ, औ	॒	॒	रामेषु	
7-3	आ	सु		रमासु	

**Table-3.11 Locative vowel ending analysis pattern**

### 3.1.7.1 Special cases/exceptions

Some exceptions are found in locative singular pronouns, for example, *sarva*. System analyzes such forms with the help of example, base as shown below-

Subanta	Analysis
सर्वस्मिन्	सर्व + डि सप्तमी एकवचन

**Table-3.12 : Locative vowel ending exception analysis pattern**

### 3.1.8 Vocative singular/dual/plural

Nothing interesting happens in the vocative. Only हे gets prefixed (with space) to the nominative forms. However, the *visarga* is omitted, as shown in the following table –

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
8-1	अ, इ, ई, उ,			हे राम, हे हरे,	नदी and all

	ऊ, औ			हे पपी, हे भानु, हे खलपू, हे ग्लौ	feminie ई ending
8-1	आ, ऋ,	।	।	हे रमा, हे लता,	विश्वपा
8-1	ऐ	।:	ॡ	हे रा	
8-2	अ, आ, इ, ई, उ, ऊ, ऋ, ए, ऐ, ओ, औ	।		हे रामौ	
8-2	आ	।	ॠ	हे रमे, विश्वपे	
8-2	इ	।	ॡ	हे हरी, हे मती	
8-2	ई	।	ॠ	हे नद्यौ	
8-2	उ	।	ॡ	हे भानू	
8-2	ऊ	।	ॡ	हे बधू	
8-2	ऋ	।	ॡ	हे पितरौ, हे धातारौ	
8-2	ओ	।	ॠ	हे गावौ	
8-2	ऐ	।	ॡ	हे रायौ	
8-2	औ	।	ॠ	हे ग्लावौ	
8-3	अ, आ,	।:	।	हे रामा:, हे रमा:	
8-3	इ	।:	ॡ	हे हरय:	
8-3	ई	।य:/:	ॠ	हे पय्य:/ हे नदी:	
8-3	उ	।व:	ॡ	हे गुख:	

8-3	ऊ	राः	रु	हे गावः	
8-3	ऋ	रः	रु	हे धातारः	
8-3	ओ	वः	वे	हे गावः	
8-3	ऐ	यः	ये	हे रायः	
8-3	औ	वः	वे	हे ग्लावः	

**Table-3.13 : Vocative vowel ending analysis pattern**

## 3.2 Consonant ending *prātipadika*

Consonant ending *prātipadika* forms are generated with modifications in the base and suffixes. Details are given below -

### 3.2.1 Nominative singular/dual/plural

System will recognize the vibhakti marker as the end character of padas. 'ट्' / 'ङ्' / 'क्' / 'ग्' / 'त्' / 'द्' are found in nominative singular. So system will search for 'ट्' / 'ङ्' / 'क्' / 'ग्' / 'त्' / 'द्' as the final portion of the given *subanta-padas* and replace it with 'ह्'. For example, - लिङ्, लिट्, अनङ्कु, अनङ्गु. In this case, the system read these *padas*, then search for 'ट्' / 'ङ्' / 'क्' / 'ग्' / 'त्' / 'द्' and replaces it with 'ह्' as shown below -

लिङ्	:	लिह् + सु [प्रथमा एकवचन ]
लिट्	:	लिह् + सु [प्रथमा एकवचन]
अनङ्कु	:	अनङ्हु + सु [प्रथमा एकवचन]
अनङ्गु	:	अनङ्हु + सु [प्रथमा एकवचन]

In case of nominative and accusative dual (1-2/2-2), *prātipadika* forms will be 'हौ' ending. For example, लिहौ. In this case, system will replace it with 'ह्' on the place on 'हौ' and add the analysis according to program with the help of sup database.

In case of nominative and accusative plural and ablative and genitive singular (1-3/2-3/5-1/6-3), the *prātipadika* forms will be ‘हः’ ending it will be replace by ‘ह्’ and give all information with the help of sup database, which is necessary. For example, लिहः

All consonant ending nominative singular/dual/plural patterns have been displayed in the following table-

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
1-1	ह, ज्, ष्, श्	ट्/ड्/क्/ग्	ह्	लिट्/लिङ्/धुक्/धुग्	
1-1	व्	ढौ		सुद्यौः	
1-1	ल्	दा्	ल्	कमल्	
1-1	म्	णा्	म्	प्रशान्	
1-1	ण्	ऋा्	ण्	सुगण्	
1-1	न्	ा्	न्	राजा	
1-1	ध्	ध्	ध्	बुध्	
1-1	द्	द्	द्	सुपाद्	
1-1	थ्	द्	थ्	अग्निमत्/अग्निमद्	
1-1	च्	ङ्	च्	प्राङ्	
1-1	त्	ात्	त्	महान्	
1-1	स्	ान्	स्	विद्वान्	
1-2	ह्	हौ	ह्	लिहौ	
1-2	व्	वौ	व्	सुदिवौ	
1-2	ल्	लौ	ल्	कमलौ	
1-2	म्	मौ	म्	प्रशामौ	
1-2	ण्	णौ	ण्	सुगणौ	

1-2	न्	ानौ	न्	राजानौ	
1-2	ध्	धौ	ध्	बुधौ	
1-2	ज्	जौ	ज्	ऋत्विजौ	
1-2	द्	दौ	द्	सुपादौ	
1-2	थ्	थौ	थ्	अग्निमथौ	
1-2	च्	ञ्चौ	च्	प्राञ्चौ	
1-2	त्	ान्तौ	त्	महान्तौ	
1-2	ष्	षौ	ष्	दधृषौ	
1-2	श्	शौ	श्	तादृशौ	
1-2	स्	ान्सौ	स्	विद्वान्सौ	
1-3	ह	हः	ह	लिहः	
1-3	व्	वः	व्	सुदिवः	
1-3	ल्	लः	ल्	कमलः	
1-3	म्	मः	म्	प्रशामः	
1-3	ण्	णः	ण्	सुगणः	
1-3	न्	ानः	न्	राजानः	
1-3	ध्	धः	ध्	बुधः	
1-3	ज्	जः	ज्	ऋत्विजः	
1-3	द्	दः	द्	सुपादः	
1-3	थ्	थः	थ्	अग्निमथः	
1-3	च्	ञ्चः	च्	प्राञ्चः	
1-3	त्	ान्तः	त्	महान्तः	
1-3	ष्	षः	ष्	दधृषः	
1-3	श्	शः	श्	तादृशः	

1-3	स्	त्सः	स्	विद्वान्सः	
-----	----	------	----	------------	--

**Table-3.14 : Nominative consonant ending analysis pattern**

### 3.2.2 Accusative singular/dual/plural

In case of accusative singular (2-1), the *prātipadika* forms will be ‘हम्’ ending ending. It will be replaced by ‘ह्’ and give other information with the help of sup database. The sup database as shown below –

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
2-1	ह्	हम्	ह्	लिहम्	
2-1	व्	वम्	व्	सुदिवम्	
2-1	ल्	लम्	ल्	कमलम्	
2-1	म्	मम्	म्	प्रशामम्	
2-1	ण्	णम्	ण्	सुगणम्	
2-1	न्	नम्	न्	राजानम्	
2-1	ध्	धम्	ध्	बुधम्	
2-1	ज्	जम्	ज्	ऋत्विजम्	
2-1	द्	दम्	द्	सुपादम्	
2-1	थ्	थम्	थ्	अग्निमथम्	
2-1	च्	ञ्चम्	च्	प्राञ्चम्	
2-1	त्	न्तम्	त्	महान्तम्	
2-1	ष्	षम्	ष्	दधृषम्	
2-1	स्	न्सम्	स्	विद्वान्सम्	



2-2	ह	हौ	ह	लिहौ	
2-2	व्	वौ	व्	सुदिवौ	
2-2	ल्	लौ	ल्	कमलौ	
2-2	म्	मौ	म्	प्रशामौ	
2-2	ण्	णौ	ण्	सुगणौ	
2-2	न्	नौ	न्	राजानौ	
2-2	ध्	धौ	ध्	बुधौ	
2-2	ज्	जौ	ज्	ऋत्विजौ	
2-2	द्	दौ	द्	सुपादौ	
2-2	थ्	थौ	थ्	अग्निमथौ	
2-2	च्	ञ्चौ	च्	प्राञ्चौ	
2-2	त्	तन्तौ	त्	महान्तौ	
2-2	ष्	षौ	ष्	दधृषौ	
2-2	श्	शौ	श्	तादृशौ	
2-2	स्	तन्सौ	स्	विद्वान्सौ	
2-3	ह	हः	ह	लिहः	
2-3	व्	वः	व्	सुदिवः	
2-3	ल्	लः	ल्	कमलः	
2-3	म्	मः	म्	प्रशामः	
2-3	ण्	णः	ण्	सुगणः	
2-3	न्	ज्ञः	न्	राज्ञः	
2-3	ध्	धः	ध्	बुधः	
2-3	ज्	जः	ज्	ऋत्विजः	
2-3	द्	दः	द्	सुपदः	

2-3	थ्	थः	थ्	अग्निमथः	
2-3	च्	ञ्चः	च्	प्राञ्चः	
2-3	त्	तः	त्	महतः	
2-3	ष्	षः	ष्	दधृषः	
2-3	श्	शः	श्	तादृशः	
2-3	स्	षः	स्	विदुषः	

**Table-3.15 : Accusative consonant ending analysis pattern**

### 3.2.3 Instrumental singular/dual/plural

In case of instrumental singular (3-1), the *prātipadika* forms will be ‘ह’ ending It will be replaced by ‘ह्’ and give all information with the help of sup database.

In case of instrumental, dative and ablative dual (3-2/4-2/5-2), *prātipadika* forms will be ‘ङ्भ्याम्’, ‘ट्भ्याम्’, ‘क्भ्याम्’, ‘ग्भ्याम्’ ending. System will isolate ‘ङ्भ्याम्’, ‘ट्भ्याम्’, ‘क्भ्याम्’, ‘ग्भ्याम्’ and search for analysis by matching in the sup database.

In case of instrumental plural (3-3), *prātipadika* forms will be ‘ङ्भिः/ट्भिः/ग्भिः/क्भिः’ ending. System will isolate ‘ङ्भिः/ट्भिः/ग्भिः/क्भिः’ and search for analysis by matching in the sup database.

All consonant ending istrumental singular/dual/plural patterns have been displayed in the following table-

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
3-1	ह्	हा	ह्	लिहा	
3-1	व्	वा	व्	सुदिवा	
3-1	ल्	ला	ल्	कमला	

3-1	म्	मा	म्	प्रशामा	
3-1	ण्	णा	ण्	सुगणा	
3-1	न्	ज्ञा	जन्	राज्ञा	
3-1	ध्	धा	ध्	बुधा	
3-1	ज्	जा	ज्	ऋत्विजा	
3-1	द्	दा	द्	सुपादा	
3-1	थ्	था	थ्	अग्निमथा	
3-1	च्	ञ्चा	च्	प्राञ्चा	
3-1	त्	ता	त्	महता	
3-1	ष्	षा	ष्	दधृषा	
3-1	श्	शा	श्	तादृशा	
3-1	स्	षा	स्	विदुषा	
3-2	ह	ङ्भ्याम्/ट्भ्याम्/ग्भ्याम्/ क्भ्याम्	ह	लिङ्भ्याम्/लिट्भ्याम्/धुग्भ्याम्	
3-2	ल्	ल्भ्याम्	ल्	कमल्भ्याम्	
3-2	म्	न्भ्याम्	म्	प्रशान्भ्याम्	
3-2	ण्	ण्भ्याम्	ण्	सुगण्भ्याम्	
3-2	न्	भ्याम्	न्	राजभ्याम्	
3-2	ध्, त्, थ्, द्, स्	द्भ्याम्	ध्	भुद्भ्याम्	
3-3	ह	ङिभः/टिभः/ग्भिः/क्भिः	ह	लिङिभः/लिटिभः/धुग्भिः	
3-3	ल्	ल्भिः	ल्	कमल्भिः	
3-3	म्	न्भिः	म्	प्रशान्भिः	
3-3	ण्	ण्भ्याम्	ण्	सुगण्भ्याम्	

3-3	न्	भिः	न्	राजभिः	
3-3	ध्, त्, थ्, द्, स्	द्धिः	ध्	भुद्धिः	

**Table-3.16 : Instrumental consonant ending analysis pattern**

### 3.2.4 Dative singular/dual/plural

In case of dative singular (4-1), the *prātipadika* forms will be ‘हे’ ending. System will isolate ‘हे’ and search for analysis by matching in the sup database.

In case of dative and ablative plural (4-3/5-3), the *prātipadika* forms will be ‘ङ्भ्यः/ट्भ्यः/ग्भ्यः/क्भ्यः’ ending. System will isolate ‘ङ्भ्यः/ट्भ्यः/ग्भ्यः/क्भ्यः’ and search for analysis by matching in the sup database.

All consonant ending dative singular/dual/plural patterns have been displayed in the following table-

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
4-1	ह्	हे	ह्	लिहे	
4-1	व्	वे	व्	सुदिवे	
4-1	ल्	ले	ल्	कमले	
4-1	म्	ने	म्	प्रशाने	
4-1	ण्	णे	ण्	सुगणे	
4-1	न्	ज्ञे	जन्	राज्ञे	
4-1	ध्	धे	ध्	बुधे	
4-1	ज्	जे	ज्	ऋत्विजे	
4-1	द्	दे	द्	सुपादे	

4-1	थ्	थे	थ्	अग्निमथे	
4-1	च्	ञ्चे	च्	प्राञ्चे	
4-1	त्	ते	त्	महते	
4-1	ष्	षे	ष्	दधृषे	
4-1	श्	शे	श्	तादृशे	
4-1	स्	से	स्	विदुसे	
4-2	ह्	ङ्भ्याम्/ट्भ्याम्/ग्भ्याम्/ क्भ्याम्	ह्	लिङ्भ्याम्/लिट्भ्याम्/धुग्भ्याम्	
4-2	ल्	ल्भ्याम्	ल्	कमल्भ्याम्	
4-2	म्	न्भ्याम्	म्	प्रशान्भ्याम्	
4-2	ण्	ण्भ्याम्	ण्	सुगण्भ्याम्	
4-2	न्	भ्याम्	न्	राजभ्याम्	
4-2	ध्, त्, थ्, द्, स्	द्भ्याम्	ध्	भुद्भ्याम्	
4-3	ह्,	ङ्भ्यः/ट्भ्यः/ग्भ्यः/क्भ्यः	ह्	लिङ्भ्यः/लिट्भ्यः/धुग्भ्यः	
4-3	ल्	भ्यः	ल्	कमल्भ्यः	
4-3	म्	न्भ्यः	म्	प्रशान्भ्यः	
4-3	ण्	ण्भ्यः	ण्	सुगण्भ्यः	
4-3	न्	भ्यः	न्	राजभ्यः	
4-3	ध्, त्, थ्, द्, स्	द्भ्यः	ध्	भुद्भ्यः	

**Table-3.17 : Dative consonant ending analysis pattern**

### 3.2.5 Ablative singular/dual/plural

Ablative singular patterns have been done according to nominative plural, dual, instrumental dual and plural and dative plural.

The consonant ending nominative singular/dual/plural patterns have been displayed in the following table -

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
5-1	ह्	हः	ह्	लिहः	
5-1	व्	वः	व्	सुदिवः	
5-1	ल्	लः	ल्	कमलः	
5-1	म्	मः	म्	प्रशामः	
5-1	ण्	णः	ण्	सुगणः	
5-1	न्	ज्ञः	न्	राज्ञः	
5-1	ध्	धः	ध्	बुधः	
5-1	ज्	जः	ज्	ऋत्विजः	
5-1	द्	दः	द्	सुपदः	
5-1	थ्	थः	थ्	अग्निमथः	
5-1	च्	ञ्चः	च्	प्राञ्चः	
5-1	त्	तः	त्	महतः	
5-1	ष्	षः	ष्	दधृषः	
5-1	श्	शः	श्	तादृशः	
5-1	स्	षः	स्	विदुषः	
5-2	ह्	ङ्भ्याम्/ट्भ्याम्/ग्भ्याम्/	ह्	लिङ्भ्याम्/लिट्भ्याम्/धुग्भ्याम्	

		क्भ्याम्			
5-2	ल्	ल्भ्याम्	ल्	कमल्भ्याम्	
5-2	म्	न्भ्याम्	म्	प्रशान्भ्याम्	
5-2	ण्	ण्भ्याम्	ण्	सुगण्भ्याम्	
5-2	न्	भ्याम्	न्	राजभ्याम्	
5-2	ध्, त्, थ्, द्, स्	द्भ्याम्	ध्	भुद्भ्याम्	
5-2	ह्,	ङ्भ्यः/ट्भ्यः/ग्भ्यः/क्भ्यः	ह्	लिङ्भ्यः/लिट्भ्यः/धुग्भ्यः	
5-2	ल्	भ्यः	ल्	कमल्भ्यः	
5-2	म्	न्भ्यः	म्	प्रशान्भ्यः	
5-2	ण्	ण्भ्यः	ण्	सुगण्भ्यः	
5-2	न्	भ्यः	न्	राजभ्यः	
5-2	ध्, त्, थ्, द्, स्	द्भ्यः	ध्	भुद्भ्यः	

**Table-3.18 : Ablative consonant ending analysis pattern**

### 3.2.6 Genitive singular/dual/plural

In case of genitive and locative dual (6-2/7-2), the *prātipadika* forms will be ‘होः’ ending. System will isolate ‘होः’ and search for analysis by matching in the sup database.

In case of genitive plural (6-3), the *prātipadika* forms will be ‘हाम्’ ending. System will isolate ‘हाम्’ and search for analysis by matching in the sup database.

The consonant ending genitive singular/dual/plural patterns have been displayed in the following table-

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
6-1	ह्	हः	ह्	लिहः	
6-1	व्	वः	व्	सुदिवः	
6-1	ल्	लः	ल्	कमलः	
6-1	म्	मः	म्	प्रशामः	
6-1	ण्	णः	ण्	सुगणः	
6-1	न्	ज्ञः	न्	राज्ञः	
6-1	ध्	धः	ध्	बुधः	
6-1	ज्	जः	ज्	ऋत्विजः	
6-1	द्	दः	द्	सुपदः	
6-1	थ्	थः	थ्	अग्निमथः	
6-1	च्	ञ्चः	च्	प्राञ्चः	
6-1	त्	तः	त्	महतः	
6-1	ष्	षः	ष्	दधृषः	
6-1	श्	शः	श्	तादृशः	
6-1	स्	षः	स्	विदुषः	
6-2	ह्	होः	ह्	लिहोः	
6-2	व्	वोः	व्	सुदिवोः	
6-2	ल्	लोः	ल्	कमलोः	
6-2	म्	मोः	म्	प्रशामोः	
6-2	ण्	णोः	ण्	सुगणोः	
6-2	न्	ज्ञोः	जन्	राज्ञोः	
6-2	ध्	धे	ध्	बुधे	



6-2	ज्	जोः	ज्	ऋत्विजोः	
6-2	थ्	थोः	थ्	अग्निमथोः	
6-2	च्	ञ्चोः	च्	प्राञ्चोः	
6-2	त्	तोः	त्	महतोः	
6-2	ष्	षोः	ष्	दधृषोः	
6-2	श्	शोः	श्	तादृशोः	
6-2	स्	सुसोः	स्	विदुसोः	
6-3	ह्	हाम्	ह्	लिहाम्	
6-3	व्	वाम्	व्	सुदिवाम्	
6-3	ल्	लाम्	ल्	कमलाम्	
6-3	म्	माम्	म्	प्रशामाम्	
6-3	ण्	णाम्	ण्	सुगणाम्	
6-3	न्	ज्ञाम्	जन्	राज्ञाम्	
6-3	ध्	धाम्	ध्	बुधाम्	
6-3	ज्	जाम्	ज्	ऋत्विजाम्	
6-3	द्	दाम्	द्	सुपदाम्	
6-3	थ्	थाम्	थ्	अग्निमथाम्	
6-3	च्	ञ्चाम्	च्	प्राञ्चाम्	
6-3	त्	ताम्	त्	महताम्	
6-3	ष्	षाम्	ष्	दधृषाम्	
6-3	श्	शाम्	श्	तादृशाम्	
6-3	स्	सुसाम्	स्	विदुषाम्	

**Table-3.19 : Genitive consonant ending analysis pattern**

### 3.2.7 Locative singular/dual/plural

In case of locative singular (7-1), the *prātipadika* forms will be ‘हे’ ending. System will isolate ‘हे’ and search for analysis by matching in the sup database.

In case of locative plural (7-3), the *prātipadika* forms will be ‘ट्सु/क्षु’ ending. System will isolate ‘ट्सु/क्षु’ and search for analysis by matching in the sup database.

The consonant ending locative singular/dual/plural patterns have been displayed in the table-

Kāraka-Vibhakti	Bases ending in	Subanta ending in	Change in Base (if any)	Examples	Exceptions /special cases
7-1	ह्	हि	ह्	लिहि	
7-1	व्	वि	व्	सुदिवि	
7-1	म्	मि	म्	प्रशामि	
7-1	ण्	णि	ण्	सुगणि	
7-1	न्	नि/ङि	न्/जन्	राजनि/राङि	
7-1	ध्	धे	ध्	बुधे	
7-1	ज्	जि	ज्	ऋत्विजि	
7-1	द्	दि	द्	सुपदि	
7-1	थ्	थि	थ्	अग्निमथि	
7-1	च्	ञ्चि	च्	प्राञ्चि	
7-1	त्	ते	त्	महते	
7-1	ष्	षोः	ष्	दधृषोः	
7-1	श्	शि	श्	तादृशि	
7-1	स्	सि	स्	विदुसि	

7-2	ह्	होः	ह	लिहोः	
7-2	व्	वोः	व्	सुदिवोः	
7-2	ल्	लोः	ल्	कमलोः	
7-2	म्	मोः	म्	प्रशामोः	
7-2	ण्	णोः	ण्	सुगणोः	
7-2	न्	ज्ञोः	जन्	राज्ञोः	
7-2	ध्	धे	ध्	बुधे	
7-2	ज्	जोः	ज्	ऋत्विजोः	
7-2	थ्	थोः	थ्	अग्निमथोः	
7-2	च्	ज्चोः	च्	प्राज्चोः	
7-2	त्	तोः	त्	महतोः	
7-2	ष्	षोः	ष्	दधृषोः	
7-2	श्	शोः	श्	तादृशोः	
7-2	स्	सुसोः	स्	विदुसोः	
7-3	ह्	दत्सु/क्षु	ह	लिट्सु/दसु/धुक्षु	
7-3	व्	exception	व्	सुद्युषु	
7-3	ल्	ल्सु	ल्	कमल्सु	
7-3	म्	न्सु/न्सु	म्	प्रशान्सु/प्रशान्सु	
7-3	ण्	ण्सु	ण्	सुगण्सु	
7-3	न्	सु	न्	राजसु	
7-3	ध्	ध्सु	ध्	बुध्सु	
7-3	ज्	क्ष्	ज्	ऋत्विक्क्ष्	
7-3	द्	दसु	द्	सुपादसु	
7-3	थ्	त्सु	थ्	अग्निमत्सु	

7-3	च्	ङ्खु/क्षु/षु	च्	प्राङ्खु/क्षु/षु	
7-3	त्	त्सु	त्	महत्सु	
7-3	ष्	क्षु	ष्	दधृक्षु	
7-3	स्	त्सु	स्	विद्वत्सु	

**Table-3.20 : Locative consonant ending analysis pattern**

### 3.2.8 Vocative singular/dual/plural

The consonant ending nominative singular/dual/plural patterns have been done according to the nominative in the following table-

<b>Kāraka-Vibhakti</b>	<b>Bases ending in</b>	<b>Subanta ending in</b>	<b>Change in Base (if any)</b>	<b>Examples</b>	<b>Exceptions /special cases</b>
8-1	ह, ज्, ष्, श्	ट्/ड्/क्/ग्	ह	हे लिट्/लिङ्/धुक्/धुग्	
8-1	व्	यौ		हे सुद्यौः	
8-1	ल्	ल्	ल्	हे कमल्	
8-1	म्	न्	म्	हे प्रशान्	
8-1	ण्	ण्	ण्	हे सुगण्	
8-1	न्	त्	न्	हे राजा	
8-1	ध्	ध्	ध्	हे बुध्	
8-1	द्	द्	द्	हे सुपाद	
8-1	थ्	द्	थ्	हे अग्निमत/अग्निमद	
8-1	च्	ङ्	च्	हे प्राङ्	

8-1	त्	ात्	त्	हे महान्	
8-1	स्	ान्	स्	हे विद्वान्	
8-2	ह	हौ	ह	हे लिहौ	
8-2	व्	वौ	व्	हे सुदिवौ	
8-2	ल्	लौ	ल्	हे कमलौ	
8-2	म्	मौ	म्	हे प्रशामौ	
8-2	ण्	णौ	ण्	हे सुगणौ	
8-2	न्	ानौ	न्	हे राजानौ	
8-2	ध्	धौ	ध्	हे बुधौ	
8-2	ज्	जौ	ज्	हे ऋत्विजौ	
8-2	द्	दौ	द्	हे सुपादौ	
8-2	थ्	थौ	थ्	हे अग्निमथौ	
8-2	च्	ञ्चौ	च्	हे प्राञ्चौ	
8-2	त्	ान्तौ	त्	हे महान्तौ	
8-2	ष्	षौ	ष्	हे दधृषौ	
8-2	श्	शौ	श्	हे तादृशौ	
8-2	स्	ान्सौ	स्	हे विद्वान्सौ	
8-3	ह	हः	ह	हे लिहः	
8-3	व्	वः	व्	हे सुदिवः	
8-3	ल्	लः	ल्	हे कमलः	
8-3	म्	मः	म्	हे प्रशामः	
8-3	ण्	णः	ण्	हे सुगणः	
8-3	न्	ानः	न्	हे राजानः	
8-3	ध्	धः	ध्	हे बुधः	

8-3	ज्	जः	ज्	हे ऋत्विजः	
8-3	द्	दः	द्	हे सुपादः	
8-3	थ्	थः	थ्	हे अग्निमथः	
8-3	च्	ञ्चः	च्	हे प्राञ्चः	
8-3	त्	तन्तः	त्	हे महान्तः	
8-3	ष्	षः	ष्	हे दधृषः	
8-3	श्	शः	श्	हे तादृशः	
8-3	स्	तन्सः	स्	हे विद्वान्सः	

**Table-3.21 : Vocative consonant ending analysis pattern**

### 3.3 Complicated *subantas* and ambiguity handling

Some *subanta* forms which are very process intensive and involve a lot of string processing have been handled with the help of example base. We have stored complicated *subanta* forms with analysis in the example base. Our database of subanta exceptions is in the following format-

Id	Subanta	Analysis
1	अहम्/अहं	अस्मद् + सु [प्रथमा एकवचन]
2	आवाम्/आवां	अस्मद् + औ/औट् [प्रथमा/द्वितीया द्विवचन]
3	वयम्/वयं	अस्मद् + जस् [प्रथमा बहुवचन]
4	माम्/मां	अस्मद् + अम् [द्वितीया एकवचन]
5	अस्मान्	अस्मद् + शस् [द्वितीया बहुवचन]
6	मया	अस्मद् + टा [तृतीया एकवचन]
7	आवाभ्याम्/आवाभ्यां	अस्मद् + भ्याम् [तृतीया/चतुर्थी/पञ्चमी द्विवचन]

8	अस्माभिः	अस्मद् + भिस् [तृतीया बहुवचन]
9	अस्मभ्यम्/अस्मभ्यं	अस्मद् + भ्यस् [चतुर्थी बहुवचन]
10	मह्यम्/मह्यं	अस्मद् + डे [चतुर्थी एकवचन]
1-1	अस्मत्	अस्मद् + भ्यस् [पञ्चमी बहुवचन]
1-2	मत	अस्मद् + डसि [पञ्चमी एकवचन]
1-3	मम	अस्मद् + डस् [षष्ठी एकवचन]
14	आवयोः	अस्मद् + ओस् [षष्ठी/सप्तमी द्विवचन]
15	अस्माकम्/अस्माकं	अस्मद् + आम् [षष्ठी बहुवचन]
16	मयि	अस्मद् + डि [सप्तमी एकवचन]
17	अस्मासु	अस्मद् + सुप् [सप्तमी बहुवचन]
18	सः	तद् + सु [प्रथमा एकवचन]
19	तौ	तद् + औ/औट् [प्रथमा/द्वितीया द्विवचन]
20	ते	तद् + जस् [प्रथमा बहुवचन]
2-1	तम्/तं	तद् + अम् [द्वितीया एकवचन]
2-2	तान्	तद् + शस् [द्वितीया बहुवचन]
2-3	तेन	तद् + टा [तृतीया एकवचन]
24	ताभ्याम्/ताभ्यां	तद् + भ्याम् [तृतीया/चतुर्थी/पञ्चमी द्विवचन]
25	तस्मै	तद् + डे [चतुर्थी एकवचन]
26	तेभ्यः	तद् + भ्यस् [चतुर्थी/पञ्चमी बहुवचन]
27	तस्मात्	तद् + डसि [पञ्चमी एकवचन]
28	तयोः	तद् + ओस् [षष्ठी/सप्तमी द्विवचन]
29	तेषाम्/तेषां	तद् + आम् [षष्ठी बहुवचन]

Table-3.22 : Example base database

### 3.3.1 Ambiguous *prātipadika*

There are two kinds of ambiguity in the *prātipadika* – those due to end-character in the base and those due to common POS (Part of Speech)

### 3.3.1.1 Ambiguity due to end-character

Some *prātipadikas* ending in a particular character combination can lead to more than one result as shown below. At this point, the system will report both the possibilities which will be reduced to one when sentential analysis is done -

लिङ् = लिह् + सु / लिश् + सु [प्रथमा एकवचन]  
लिट् = लिह् + सु / लिश् + सु [प्रथमा एकवचन]  
विट् = विह् + सु / विश् + सु [प्रथमा एकवचन]  
विङ् = विह् + सु / विश् + सु [प्रथमा एकवचन]  
तादृक् = तादृश् + सु / तादृह् + सु [प्रथमा एकवचन]  
तादृग् = तादृश् + सु / तादृह् + सु [प्रथमा एकवचन]

### 3.3.1.2 Morphological ambiguity (categorical - verb/noun)

Many *subantas* (mostly the *śatr pratyayānta* in locative singular) can be verbs as well as nouns. Therefore, they will be picked up as verb and not sent for *subanta* processing. For example, रामः can be a *subanta* (रामः = राम + सु प्रथमा एकवचन) as well as a verb (रामः = रम् + लट्लकार उत्तम पुरुष बहुवचन). At this point, the verb database excludes such ambiguous cases.

### 3.3.2 Ambiguous *vibhaktis*

Same forms are available in the dual of nominative and accusative cases like- रामौ, dual of instrumental, dative and ablative cases like- रामाभ्याम्, plural of dative and ablative cases like- रामेभ्यः, and dual of relative and locative cases like - रामयोः. In neuter gender as well, the nominative and accusative singular forms may be identical as in पुस्तकम् (1-1 and 2-1). In such cases, the system will give all possible results as in

रामौ = औ/औट् [1.2&2.2]



रामाभ्याम्	=	भ्याम्	[3.2, 4.2&5.2]
रामेभ्यः	=	भ्यस्	[4.3&5.3]
रामयोः	=	ओस्	[6.2 & 7.2]
पुस्तकम्	=	सु/अम्	[1.1 & 2.1]
हरेः	=	डसि/डस्	[5.1 & 6.1]

### 3.3.2.1 Always ambiguous *vibhaktis* (औ/औट्, भ्याम्, भ्यस्, ओस् etc.)

This is a very complicated problem for machine and can not be handled at sub-sentential level as in the case of *subantas*. The system will give multiple results in these cases.

रामौ	=	औ/औट्	[1.2&2.2]
रामाभ्याम्	=	भ्याम्	[3.2, 4.2&5.2]
रामेभ्यः	=	भ्यस्	[4.3&5.3]
रामयोः	=	ओस्	[6.2 & 7.2]

## **Chapter-IV**

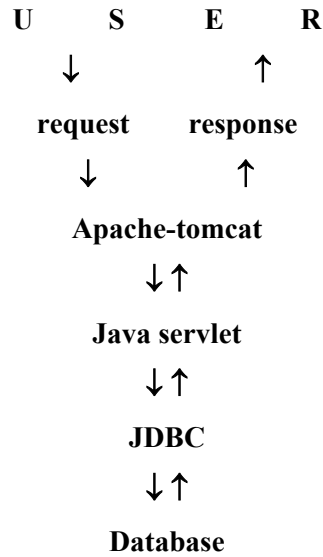
### **ONLINE SUBANTA RECOGNIZER AND ANALYZER**

## 4.1 Description of SRAS

The present system model uses Java in the web format for the recognition and analysis (*prakṛti-pratyaya vibhāga*) of *subanta-padas* from Sanskrit texts according to Pāṇinian and *Siddhānta Kaumudī* (SK) formalism. The system accepts words/sentences/ text Devanagari utf-8 input in the text area and gives analyzed output in Devanagari utf-8 format. Recognition of *subanta-padas* takes place first according to the process as outlined in chapter-II. The analysis follows according to the processes as outlined in chapter-III. The recognition process includes identifying all non-subanta categories first. This is as a by-product of this system get preliminary POS tagged for punctuations, *avyayas* and verbs. At this point, the system does not give a facility for uploading and spooling multiple files, but the next version is going to incorporate this and many more features. The SRAS has twin goals in the mind – simplification of Sanskrit texts for self reading and M(A)T.

### 4.1.1 Architecture of the system

The following model describes the interaction between multi-tiered architecture of the SRAS



#### 4.1.1.1 Multi-layered architecture

The architecture of the SRAS is multilayered java webserver environment, with front end, database connectivity and backend.

#### 4.1.1.2 The front-end: online interface

The Graphical User Interface (GUI) of the SRAS, which is seen by users, is its front-end. It is produced by JSP (Java Server Pages) and HTML components. The main JSP file rsubanta.jsp allows the user to give input in Devanagari utf-8 format using HTML text area component. Upon clicking the button labeled [“सुबन्त पहचान एवं प्रकृति-प्रत्यय विभाग के लिये यहां क्लिक करें”](#), it calls the reverse subanta Java object (called RSubanta) for processing the input. The output as given by the Java objects is displayed to the user in Devanagari utf-8 format.

The screenshot shows a web interface with a title "सङ्गणक द्वारा सुबन्त पहचान और प्रकृति-प्रत्यय विभाग" (Computer-based Subanta Identification and Part-of-Speech Division). Below the title is a link: "यह कैसे कार्य करता है देवनागरी लेखन की सहायता के लिये सिस्टम की सीमायें सम्पर्क करें". A green instruction line reads: "सुबन्त पहचान एवं प्रकृति-प्रत्यय विभाग के लिये कृपया संस्कृत वाक्य, गद्य या संस्कृत पद लिखें". There is a large text area for input. Below it is a button labeled "प्रकृति-प्रत्यय विभाग के लिये यहाँ क्लिक करें" and a checkbox labeled "Run in debug mode". At the bottom, there is a section titled "परिणाम" (Result).

Figure 4.1

#### 4.1.1.3 The back-end: database / txt files

There are two versions of the system; the server based version connects to a MSSQL Server 2005 RDBMS through JDBC. The rule base, example base and other linguistic resources are stored as Devanagari utf-8. The PC based portable version, for obvious reasons, can not have RDBMS support. Therefore, we have our rules and data stored in utf-8 text files as backend. A design of the reverse subanta database as give below:

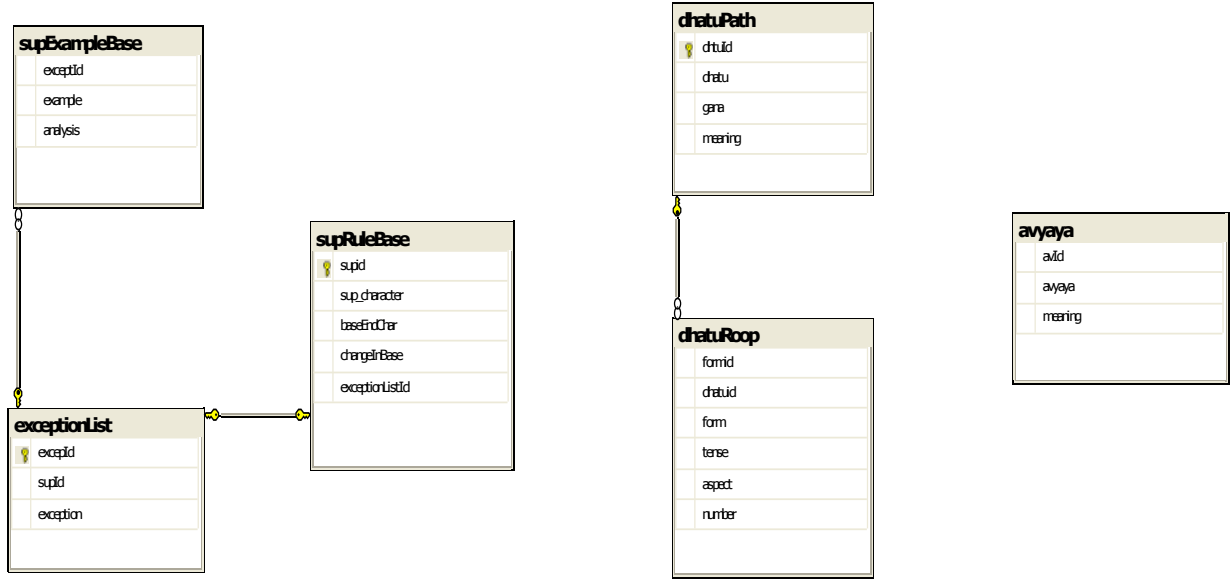


Figure 4.2

The *supRuleBase* table has relations with the *exceptionList* table. Any exception figuring in the rule base must have a description in the exception list. The table *supExampleBase* depends on the *exceptionList* and must provide analysis for each example figuring in the *exceptionList* and marked in the *supRuleBase*

The *dhāturūp* object depends on the *dhātupāṭh* object while the *avyaya* is a floating object as of now. These linguistic resources are checked for recognition of *subantas*, and the rule base-example base are searched for analysis

## Text Files

For the standalone version (PC based), the data is stored in the following text files –

- sup\_EB.txt (the subanta example base data) in the following format

अहम्=अस्मद+सु प्रथमा एकवचन;अहं=अस्मद+सु प्रथमा एकवचन;आवाम्=अस्मद+औ प्रथमा द्वितीया द्विवचन;आवां=अस्मद+औ प्रथमा द्वितीया द्विवचन;वयम्=अस्मद+जस् प्रथमा बहुवचन;वयं=अस्मद+जस् प्रथमा बहुवचन;माम्=अस्मद+अम् द्वितीया एकवचन;मां=अस्मद+अम् द्वितीया एकवचन;अस्मान्=अस्मद+शस् द्वितीया बहुवचन;मया=अस्मद+टा तृतीया एकवचन;आवाभ्यां=अस्मद+भ्याम् तृतीया, चतुर्थी, पञ्चमी द्विवचन;आवाभ्याम्=अस्मद+भ्याम् तृतीया, चतुर्थी, पञ्चमी द्विवचन;अस्माभिः=अस्मद+भिस तृतीया बहुवचन;मह्यम्=अस्मद+ङे चतुर्थी एकवचन;मह्यं=अस्मद+ङे चतुर्थी एकवचन;अस्मभ्यम्=अस्मद+भ्यस् चतुर्थी बहुवचन;अस्मभ्यं=अस्मद+भ्यस् चतुर्थी बहुवचन;मत=अस्मद+डसि पञ्चमी एकवचन;अस्मत्=अस्मद+भ्यस् पञ्चमी बहुवचन;मम=अस्मद+डस् षष्ठी एकवचन;आवयोः=अस्मद+ओस् षष्ठी, सप्तमी द्विवचन;अस्माकम्=अस्मद+आम् षष्ठी एकवचन;अस्माकं=अस्मद+आम् षष्ठी एकवचन;मयि=अस्मद+ङि सप्तमी एकवचन;अस्मासु=अस्मद+सुप् सप्तमी बहुवचन;

- sup\_RB.txt (the subanta rule base) in the following format

ॠ=ॠ+सु प्रथमा एकवचन;ॠभ्याम्=+भ्याम् तृतीया चतुर्थी पञ्चमी द्विवचन;ॠभ्यां=+भ्याम् तृतीया चतुर्थी पञ्चमी द्विवचन;ॠभ्याम्=+भ्याम् तृतीया चतुर्थी पञ्चमी द्विवचन;ॠभ्यां=+भ्याम् तृतीया चतुर्थी पञ्चमी द्विवचन;ॠभ्यः=+भ्यस् चतुर्थी पञ्चमी बहुवचन;ॠभ्यः=+भ्यस् चतुर्थी पञ्चमी बहुवचन;ॠनाम्=+आम् षष्ठी बहुवचन;ॠनां=+आम् षष्ठी बहुवचन;ॠणाम्=+आम् षष्ठी बहुवचन;ॠणां=+आम् षष्ठी बहुवचन;ॠनाम्=+आम् षष्ठी बहुवचन;ॠनाम्=+आम् षष्ठी बहुवचन;ॠणाम्=+आम् षष्ठी बहुवचन;ॠणां=+आम् षष्ठी बहुवचन;ॠस्य=+स्य डस्, षष्ठी एकवचन;ॠ=+ङि सप्तमी एकवचन;ॠ=ॠ+सु प्रथमा एकवचन;ॠ+सु प्रथमा द्वितीया द्विवचन वचन;ॠण=+टा तृतीया एकवचन;ॠ=+औ/औट् प्रथमा/द्वितीया द्विवचन;ॠः=+जस् प्रथमा बहुवचन;ॠणि=+जस् प्रथमा द्वितीया बहुवचन;ॠनि=+जस् प्रथमा द्वितीया बहुवचन;ॠम्=+अम् द्वितीया एकवचन;ॠ=+अम् द्वितीया एकवचन;ॠन्=+शस् द्वितीया बहुवचन;ॠण=+टा तृतीया एकवचन;ॠन=+टा तृतीया एकवचन;ॠः=+भिस् तृतीया बहुवचन;ॠय=+ङे चतुर्थी एकवचन;ॠभ्यः=+भ्यस् चतुर्थी/पञ्चमी बहुवचन;ॠत्=+डसि पञ्चमी एकवचन;ॠद्=+डसि पञ्चमी एकवचन;ॠस्य=+डस् षष्ठी एकवचन;ॠयोः=+ओस् षष्ठी/सप्तमी द्विवचन;ॠनां=+आम् षष्ठी बहुवचन;ॠनाम्=+आम् षष्ठी बहुवचन;ॠणां=+आम् षष्ठी बहुवचन;ॠणाम्=+आम् षष्ठी बहुवचन;ॠ=+ङि सप्तमी एकवचन;ॠषु=+सुप् सप्तमी बहुवचन;ॠसु=+सुप् सप्तमी बहुवचन;

- verb.txt (the verb form database for subanta recognition)

भवति, भवतः, भवन्ति, भवसि, भवथः, भवथ, भवामि, भवावः, भवामः, भवतु, भवताम्, भवन्तु, भव, भवतम्, भवत, भवानि, भवाव, भवाम, अभवत्, अभवताम्, अभवन्, अभवः, अभवतम्, अभवत, अभवम्, अभवाव, अभवाम, भवेत्, भवेताम्, भवेयुः, भवेः, भव

भवेत्, भवेयम्, भवेव, भवेम, बभूव, बभूवतुः, बभूवुः, बभूविथ, बभूवथुः, बभूव, बभूव, बभूविव, बभूविम, भविता, भवितारौ, भवितारः, भवितासि, भवितास्थः, भवितास्थ, भवितास्मि, भवितास्वः, भवितास्मः, भविष्यति, भविष्यतः, भविष्यन्ति, भविष्यसि, भविष्यथः, भविष्यथ, भविष्यामि, भविष्यावः, भविष्यामः, भूयात्, भूयास्ताम्, भूयासुः, भूयाः, भूयास्तम्, भूयास्त, भूयासम्, भूयास्व, भूयास्म, अभूत्, अभूताम्, अभूवन्, अभूः, अभूतम्, अभूत, अभूवम्, अभूव, अभूम, अभविष्यत्, अभविष्यताम्, अभविष्यन्, अभविष्यः, अभविष्यतम्, अभविष्यत, अभविष्यम्, अभविष्याव, अभविष्याम,

- avyaya.txt (the vyaya database for subanta recognition)

अ, कश्चित्, सदैव, अकस्मात्, अकाण्डे, अग्निसात्, अग्नी, अघोः, अङ्ग, अजस्रम्, अञ्जसा, अतः, अति, अतीव, अत्र, अथ, अथकिम्, अथवा, अथो, अब्धा, अद्य, अद्यापि, अधरात्, अधरेद्युः, अधरेण, अधः, अधस्तात्, अधि, अधिहरि, अधुना, अधोऽधः, अध्यट्, अनतः, अनिशम्, अनु, अनेकधा, अनेकशः, अन्तः, अन्तरा, अन्तरेण, अन्यतः, अन्यत्, अन्यत्र, अन्यथा, अन्यदा, अन्येद्युः, अन्वक्, अप, अपरेद्युः, अपलुपम्, अपि, अपिवा, अभि, अभितः, अभीक्ष्णम्, अमा, अमुत्र, अम्, अयि, अये, अरम्, अरे, अरेरे, अर्जुनतः, अलम्, अल्पशः, अवगाहे, अवचक्षे, अवदत्तम्, अवरतः, अवः, अवश्यम्, अव्यथिष्यै, अष्टधा, असकृत्, असाम्प्रतम्, अस्तु, अह, अहह, अहो, अह्नाय, अअ, आण, आतः, आतृदः, आदह, आदितः, आम्, आरात्, आर्यहलम्, आविः, आः, आहुवध्यै, आहो, आहोस्वित्, इ, इतरेद्युः, इतः, इति, इत्, इत्थम्, इदानीम्, इब्धा, इव, इह, ई, ईषत्, उ, उच्चैः, उत्, उताहो, उत्तरतः,

#### 4.1.1.4 Database connectivity

The database connectivity is done through JDBC driver software downloadable from the Microsoft site. The JDBC-ODBC bridge driver from Sun Mini system freely downloadable from Microsoft is not Unicode compliant. Therefore there were problems in accepting Unicode queries and displaying Unicode results. JDBC Application Programming Interface (API) is the industry standard for database independent connectivity for Java and a wide range of database-SQL databases. JDBC technology allows to use the Java programming language to develop ‘Write once, run anywhere’ capabilities for applications that require access to large scale data. JDBC works as bridge between Java program and Database. SQL server 2005 and JDBC support input and output in unicode, so this system accepts unicode devanagri text as well as prints result in unicode devanagri also<sup>232</sup>.

#### 4.1.1.5 The web server

The SRAS runs on Apache Tomcat 4.0 platform. The details for this Java based webserver follows -

<sup>232</sup> <http://java.sun.com/products/servlet/>

#### 4.1.1.5.1 Apache Tomcat 4.0

Apache Tomcat is the servlet container that is used for the Java Servlet and JavaServer Pages technologies. The Java Servlet and Java Server Pages specifications are developed by Sun under the Java Community Process. Apache Tomcat is developed in an open and participatory environment and released under the Apache Software License. Apache Tomcat is intended to be a collaboration of the best-of-breed developers from around the world<sup>233</sup>.

#### 4.1.1.5.2 Java Servlet Technology

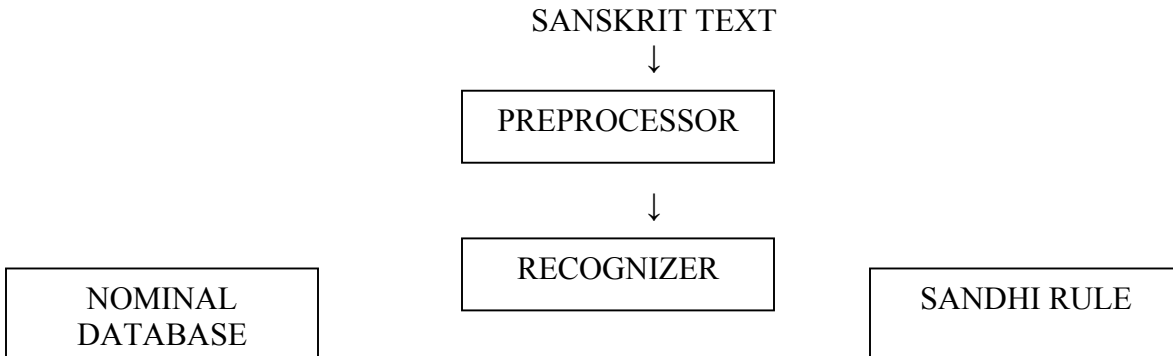
*Java Servlet technology provides web developers with a simple, consistent mechanism for extending the functionality of a web server and for accessing existing business systems. A servlet can almost be thought of as an applet that runs on the server side--without a face. Java servlets make many web applications possible<sup>234</sup>.*

#### 4.1.1.5.3 Java Server Pages

*Java Server Pages (JSP) technology provides a simplified, fast way to create dynamic web content. JSP technology enables rapid development of web-based applications that are server and platform-independent<sup>235</sup>. It is one of the most sophisticated tools available for high performance and secures web applications.*

### 4.1.2 Module Description

The process flow of the SRAS is given below -

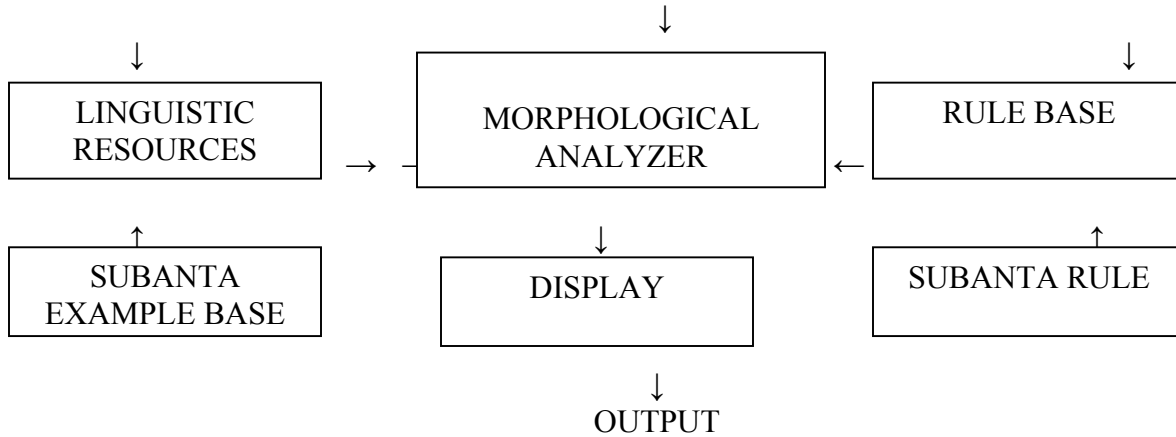


<sup>233</sup> <http://www.apache.org/>

<sup>234</sup> <http://java.sun.com/products/servlet/>

<sup>235</sup> <http://java.sun.com/products/jsp/>





Present system has three major components –

- RSubanta
- Preprocessor-Recognizer
- Analyzer

#### 4.1.2.1 RSubanta

This is the main class which tokenizes the input text, gets it preprocessed, gets subantas marked and then sends the subanta padas to the analyzer for analysis. Finally, this module displays the results. This object has following methods –

```

public String processSup(String s)
private String preProcess(String tkn)
private String analyze(String tk)
public String printErr()

```

The code sample for the processSup() function follows –

```

public String processSup(String s){
    String tkn="";
    String ts = "";

```

```

s = s.trim();
String tmp = "";
if (s.length()>0){
StringTokenizer st = new StringTokenizer(s, " ");
StringTokenizer st2 = null;

while(st.hasMoreTokens()){
    tkn = st.nextToken().trim();           tkn=

preProcess(tkn);
    if (tkn.indexOf("SUBANTA")>-1 && tkn.indexOf("PUNCT")>-1){

        st2 = new StringTokenizer(tkn, " ");
        while(st2.hasMoreTokens()){

            tmp = st2.nextToken();           if
            (tmp.indexOf("SUBANTA")>0)

            ts = ts + " "+analyze(tmp);
            else                               ts = ts
            + " "+tmp;
        }
    }
    else if (tkn.indexOf("SUBANTA")>-1)

    ts = ts+" "+analyze(tkn);

    else ts = ts + " "+tkn;

}
return "{ "+ts+" }";

}

else return "Please enter some Sanskrit text to process";

```

}

#### **4.1.2.1.1 Preprocessor / Recognizer**

This module first normalizes the input and then checks if there are subanta constituents to process. If a subanta is found, it is sent to the analyzer for analysis; else it is labeled appropriately as \_VERB or \_AVYAYA and reported back as output

#### **4.1.2.1.2 Punctuation checker**

This component checks punctuation in input text and tags with specific tag. If any punctuation is found between two characters without space “ ” then it deletes them. And if any punctuation comes to the left or right of a word without space, then this program adds a space between the word and the punctuation. For example - रामः अकथयत् यत् “ सः गृहम् गच्छतु”। this component does as रामः अकथयत् यत् [“\_PUNCT] सः गृहम् गच्छतु [”\_PUNCT] [।\_PUNCT] and send for next process

## सङ्गणक द्वारा सुबन्त पहचान और प्रकृति-प्रत्यय विभाग

यह कैसे कार्य करता है देवनागरी लेखन की सहायता के लिये [सिस्टम की सीमायें सम्पर्क करें](#)

सुबन्त पहचान एवं प्रकृति-प्रत्यय विभाग के लिये कृपया संस्कृत वाक्य, गद्य या संस्कृत पद लिखें

```
,. . , / ' ; [ ] [ = - \ 0 9 ~ 1 0 # $ 5 6 7 & & ^ % $ # & * ( ) _ _ + | ] [
( ) " : , . . , / ' ; [ ] [ = - \ 0 9 ~ 1 0 # $ 5 6 7 & & ^ % $ # & * ( ) _ _ + | ] [
( ) " : , . . , / ' ; [ ] [ = - \ 0 9 ~ 1 0 # $ 5 6 7 & & ^ % $ # & * ( ) _ _ + | ] [
( ) " : , . . , / ' ; [ ] [ = - \ 0 9 ~ 1 0 # $ 5 6 7 & & ^ % $ # & * ( ) _ _ + | ] [
( ) " :
```

प्रकृति-प्रत्यय विभाग के लिये यहाँ क्लिक करें

Run in debug mode ☐

### परिणाम

```
{ _PUNCT [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [-_PUNCT] [-_PUNCT]
[_PUNCT] [0_NUMERAL] [9_NUMERAL] [-_PUNCT] [1_NUMERAL] [@_PUNCT] [#_PUNCT] [$_PUNCT] [5_NUMERAL] [6_NUMERAL]
[7_NUMERAL] [&_PUNCT] [&_PUNCT] [^_PUNCT] [%_PUNCT] [$_PUNCT] [#_PUNCT] [&_PUNCT] [*_PUNCT] [_PUNCT] [D_PUNCT]
[_PUNCT] [_PUNCT] [+_PUNCT] [_PUNCT] [{_PUNCT} [_PUNCT] ["_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT]
[_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [-_PUNCT] [-_PUNCT] [_PUNCT]
[0_NUMERAL] [9_NUMERAL] [-_PUNCT] [1_NUMERAL] [@_PUNCT] [#_PUNCT] [$_PUNCT] [5_NUMERAL] [6_NUMERAL]
[7_NUMERAL] [&_PUNCT] [&_PUNCT] [^_PUNCT] [%_PUNCT] [$_PUNCT] [#_PUNCT] [&_PUNCT] [*_PUNCT] [_PUNCT] [D_PUNCT]
[_PUNCT] [_PUNCT] [+_PUNCT] [_PUNCT] [{_PUNCT} [_PUNCT] ["_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT]
[_PUNCT] [_PUNCT] [_PUNCT] [-_PUNCT] [-_PUNCT] [_PUNCT] [0_NUMERAL] [9_NUMERAL] [-_PUNCT] [1_NUMERAL]
[@_PUNCT] [#_PUNCT] [$_PUNCT] [5_NUMERAL] [6_NUMERAL] [7_NUMERAL] [&_PUNCT] [&_PUNCT] [^_PUNCT] [%_PUNCT]
[$_PUNCT] [#_PUNCT] [&_PUNCT] [*_PUNCT] [_PUNCT] [D_PUNCT] [_PUNCT] [_PUNCT] [+_PUNCT] [_PUNCT] [{_PUNCT} [_PUNCT]
[_PUNCT] ["_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [_PUNCT] [-_PUNCT] [-_PUNCT]
[_PUNCT] [0_NUMERAL] [9_NUMERAL] [-_PUNCT] [1_NUMERAL] [@_PUNCT] [#_PUNCT] [$_PUNCT] [5_NUMERAL] [6_NUMERAL]
[7_NUMERAL] [&_PUNCT] [&_PUNCT] [^_PUNCT] [%_PUNCT] [$_PUNCT] [#_PUNCT] [&_PUNCT] [*_PUNCT] [_PUNCT] [D_PUNCT]
[_PUNCT] [_PUNCT] [+_PUNCT] [_PUNCT] [{_PUNCT} [_PUNCT] ["_PUNCT] [_PUNCT] }
```

Figure 4.3

### Code sample

```

public class Preprocessor{
    String err = "";
    String line="";
    StringBuffer avyaya = null;
    StringBuffer verbs = null;
    public Preprocessor(StringBuffer a, StringBuffer b){
        avyaya = a;
        verbs = b;
    }
    private String checkPunct(String tkn){
        String punctProper = ",.'()[]{}$#@!%^&*-_+=|\\?/<>~`.;\"u0964 \u0965";
        String punctRoman= "abcdefghijklmnopqrstuvwxyz
                                ABCDEFGHIJKLMNOPQRSTUVWXYZ";
        String punctNum = "0123456789\u0966 \u0967 \u0968 \u0969 \u096A \u096B \u096C \u096D
\u096E \u096F";
        String sub_punct_left = "([{*`\"";
        String sub_punct_right = ")]!*`\";?.\u0964 \u0965";
        boolean av_vCheck = false;
        String punct = punctProper + punctRoman + punctNum;
        String ts = "";
        char c = ' ';
        int idx = -1;

```

#### 4.1.2.1.3 Non subanta checker

This component checks for avyayas and verbs in the non punctuation word of the input. First avyayas are checked and tagged, and then the verbs are checked and tagged.

#### Code snippet

```

private String checkNS(String tkn, String type){
    boolean found = false;
    StringTokenizer st = null;
    String ts = "";
    String av_v = "";

```

```

StringBuffer line = null;
        if (type.equals("av"))
            line = avyaya;
    else
        line = verbs;
    if (line.indexOf(tkn) > -1){
        ts = tkn;
        if ( type.equals("av") )
            ts = "["+ts+"_AV]";
    else
        ts = "["+ts+"_VERB]";
    }
    return ts;
}

```

#### 4.1.2.1.4 *Subanta* recognizer

The remaining words are tagged `_SUBANTA` by this component. Sample of this function-

```

else if (type.equals("verb")) // all untagged words, which is tagged with subanta by
this
        faction.
        ts = " "+tkn+"_SUBANTA";
    else
        ts = tkn; ]

```

#### 4.1.2.2 *Subanta* Analyzer

After the preprocessing and subanta recognition process, the `RSubanta` object sends a *subanta-pada* to the `SupAnalyzer` object for obtaining analysis. This component returns analysis of *subanta-padas* with the help of database and text files

This object has following methods –

```

public String analyzeSup(String tkn)

```

```

private String checkExampleBase(String tkn)
private String checkRuleBase(String tkn)
public String printErr()

```

SupAnalyzer first checks for solutions in the example base data file or table. If found, it does not check the rule base. If not found, it proceeds to check the rule base by splitting certain number of last characters from the *subanta-pada*. It first splits 5, then 4 and likewise up to the last character. Each time it checks if the base and affix combination it arrives at is valid or not by checking the affix in the rule base. If found, it assumes the analysis to be correct and reports back. If invalid, it proceeds to create another combination of base + affix.

Code sample from the checkExampleBase() function –

```

if ( exmpls.indexOf(";"+newTkn)>-1 ) {
    try{
        data_tkn    =    exmpls.substring(    exmpls.indexOf(";"+newTkn)+1,    exmpls.indexOf(";";
exmpls.indexOf(newTkn) ));
        if (data_tkn.indexOf("=")>0){
            ky = data_tkn.substring(0,data_tkn.indexOf("="));
            val = data_tkn.substring(data_tkn.indexOf("=")+1, data_tkn.length() );
            ky = ky.trim();
        }
        err = err+"ky="+ky+" val = "+val+"<br>";
    }
    catch (Exception ex){
        ky = "not found";
        val="not found";
        System.out.println("in EB data_tkn="+data_tkn+" ky="+ky+" val="+val+"");
    }
    if (newTkn.equals(ky)) {
        ts = newTkn+" ["+val+"] ";
    }
    else ts = tkn;
}
else

```

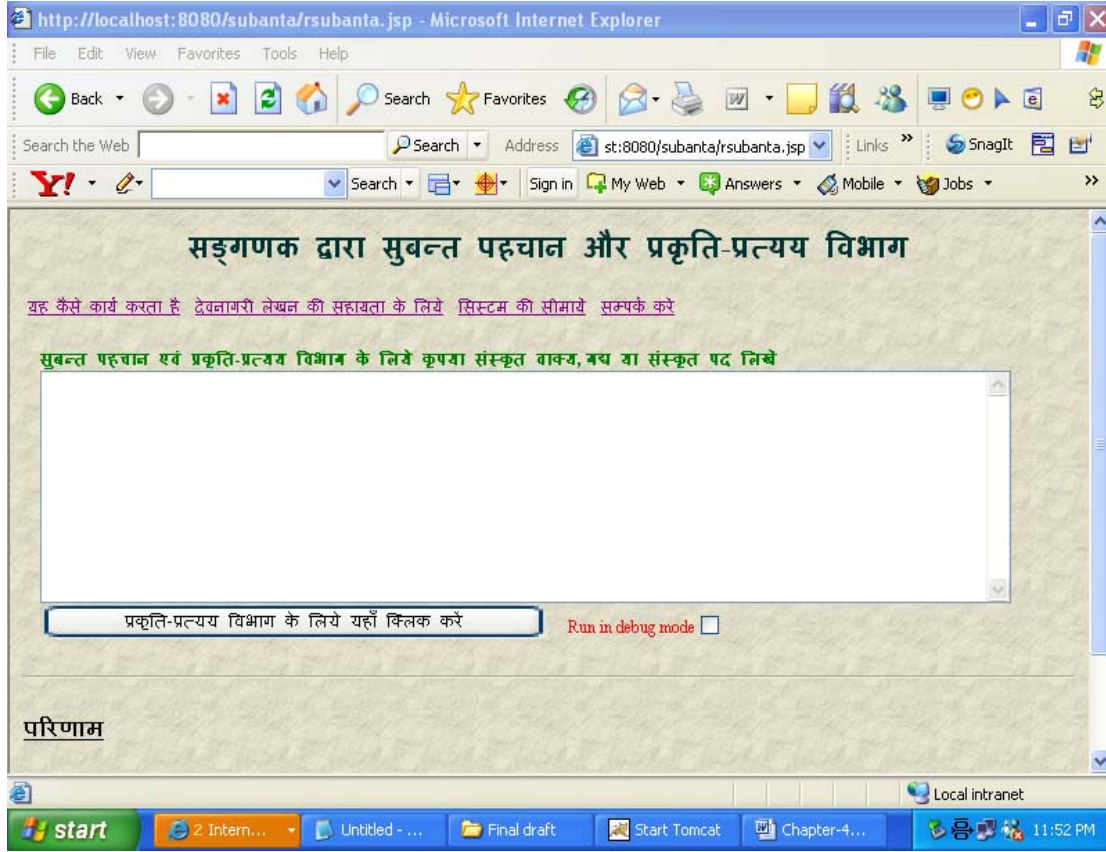
```
ts = checkRuleBase(tkn);
```

## 4.2 Test corpora

We have collected 140 files in unicode devanagari for testing this system and another proposes. Out of which the following 10 have been used in the result analysis (please see Introduction page-12 for details).

## 4.3. How it works

On the localhost (CD version), the website can be opened by the URL <http://localhost:8080/subanta/rsubanta.jsp>. On the actual server, the URL can be something like <http://www.sanskrit.jnu.ac.in/subanta> (the actual site is not live yet due to infrastructural bottlenecks). The home page of the site looks like –





**Figure 4.4**

The site accepts devanagari data in utf-8 format. Therefore, a Unicode IME like Baraha<sup>236</sup> has to be installed. Otherwise, user can enter some the test files provide.

Upon clicking the button labeled ‘देवनागरी-लेखन सहायता के लिये यहां क्लिक करें’ The JSP interface sends data to the RSubanta object, which after tokenizing the input sends each word for preprocessing and the result of preprocessor to SupAnalyzer if the word is tagged \_SUBANTA. The RSubanta keeps on building the display depending on the output from the proprocessor-recognizer and analyzer objects.

### **Input-Output examples -**

#### **Input text**

दुर्गापूजा हिन्दूनां महोत्सवः अस्ति । शरत्काले आश्विनमासे इयं पूजा सम्पूर्णे भारते श्रद्धया आनन्देन च सम्पादिता भवति । दुर्गापूजायाः विषये अनेकाः कथाः प्रचलिताः सन्ति । रावणेन सह युद्धात् पूर्वं श्रीरामः दशदिनानि यावत् दुर्गायाः पूजां कृतवान् । तस्याः प्रसादेन सः रावणं हतवान् लङ्कायां विजयं प्राप्तवान् च । तस्य विजयस्य स्मरणाय प्रतिवर्षम् इयं दुर्गापूजा अनुष्ठिता भवति

#### **Analysis of subanta-padas**

दुर्गापूजा [दुर्गापूजा+सु प्रथमा एकवचन] हिन्दूनां [हिन्दू+आम् षष्ठी बहुवचन] महोत्सवः [महोत्सव+सु प्रथमा एकवचन] [अस्ति\_VERB] [।\_PUNCT] शरत्काले [शरत्काल+डे सप्तमी एकवचन] आश्विनमासे [आश्विनमासे+डे सप्तमी एकवचन] इयं [इदम्+सु प्रथमा एकवचन] पूजा [पूजा+ सु प्रथमा एकवचन] सम्पूर्णे [सम्पूर्ण+ङि सप्तमी एकवचन] भारते [भारत+ङि सप्तमी एकवचन] श्रद्धया [श्रद्धा+टा तृतीया एकवचन] आनन्देन [आनन्द+टा तृतीया एकवचन] [च\_AV] सम्पादिता [सम्पादिता+ सु प्रथमा एकवचन] [भवति\_VERB] दुर्गापूजायाः [दुर्गापूजा+ङसि/ङस् पञ्चमी/षष्ठी एकवचन] विषये [विषय+ डे सप्तमी एकवचन] अनेकाः [अनेक+जस् प्रथमा बहुवचन] [कथाः [कथा+जस् प्रथमा बहुवचन] प्रचलिताः [प्रचलित+जस् प्रथमा बहुवचन] [सन्ति\_VERB] [।\_PUNCT] रावणेन [रावण+टा तृतीया एकवचन] [सह\_AV] युद्धात् [युद्ध+ङसि पञ्चमी एकवचन] पूर्वं [पूर्व+अम् द्वितीया एकवचन] श्रीरामः [श्रीराम+सु प्रथमा एकवचन] दशदिनानि [दशदिन+जस् प्रथमा द्वितीया बहुवचन] [यावत्\_AV] दुर्गायाः [दुर्गा+ङसि/ङस्

<sup>236</sup> <http://www.baraha.com/BarahaIME.htm>

पञ्चमी/षष्ठी एकवचन] पूजां [पूज्+आम् षष्ठी बहुवचन] कृतवान् [कृतवत्+सु प्रथमा एकवचन]  
 [।\_PUNCT] तस्याः [तद्+ङसि/ङस् पञ्चमी, षष्ठी एकवचन] प्रसादेन [प्रसाद+टा तृतीया एकवचन] सः  
 [तद्+सु प्रथमा एकवचन] रावणं [रावण+अम् द्वितीया एकवचन] हतवान् [हतवत्+ सु प्रथमा एकवचन]  
 लङ्कायां [लङ्का+ङि सप्तमी एकवचन] विजयं [विजय+अम् द्वितीया एकवचन] प्राप्तवान् [प्राप्तवत्+ सु  
 प्रथमा एकवचन] [च\_AV] [।\_PUNCT] तस्य [तद्+ङस् षष्ठी एकवचन] विजयस्य [विजय+स्य ङस्, षष्ठी  
 एकवचन] स्मरणाय [स्मरण+ङे चतुर्थी एकवचन] प्रतिवर्षम् [प्रतिवर्ष+अम् द्वितीया एकवचन] इयं [इदम्+सु  
 प्रथमा एकवचन] दुर्गापूजा [दुर्गापूज्+सु प्रथमा एकवचन] अनुष्ठिता [अनुष्ठिता+सु प्रथमा एकवचन]  
 [भवति\_VERB] [।\_PUNCT] }

### 4.3.1 Result analysis and limitation

System prints result as output in three colors - Black, Blue and Red. All analyzed *subanta* in black color, non *subanta* (punctuations /avyaya/verb) in blue color and all un-analyzed *subanta* in red color. The result has three structure, *subanta-pada* [analysis] in Black color [non-subanta with tag] in Blue color and unanalyzed *subanta* with '\_SUBANTA' tag in Red color.

Currently this system is giving multiple results in ambiguous cases, which will be solved by analyzing the context at the sentence level. A doctoral work to analyze karaka mechanically is nearing its completion. The feedback from the karaka component will be taken to disambiguate between multiple results in this case. In addition, we will be adding some constraints in the forms of additional rules or data to minimize ambiguity. At present, we have only 90,000 primary verb forms in the verb database, which are commonly found in Sanskrit literature. Though it is very unlikely that ordinary Sanskrit literature will overshoot this list, yet the system is likely to start processing a verb as *subanta* if not found in the database. We have 519 *avyayas* listed in our *avyaya* database. it is not enough for *avyaya* recognition in ordinary Sanskrit literature. In this case, the system is likely to start processing an *avyaya* as *subanta*, if it is not found in *avyaya* database. We will try to improve results and database.

## **CONCLUSION**

The present R & D is an attempt to process Sanskrit NP inflections by way of Pāṇini's rule system, appropriate database and example-base. The system developed is an online system run on Apache Tomcat platform using Java servlet and MSSQL server 2005 as back end. The thesis includes a PC version run on Apache-Tomcat and works on data in text files. The goal is to simplify Sanskrit text for self reading, understanding, and also for any Machine (Aided) Translation from Sanskrit to other languages.

## Limitations of the system

This system has been developed according to Pāṇinian formulation. System accepts only non-joint (*sandhi-rahita*) Sanskrit text in *devanagari* script. This system fully depends on both the rule base, example base and a database of other linguistic resources. It recognizes *subanta* by marking all non-subanta in the sentence by checking the verb and avyaya database. At this point, the system takes Sanskrit text in utf-8 format (in a text area), the future upgrade will include a feature by which multiple files can be uploaded, queued for processing and will E-mail the processed files to the sender. Since it is an online system, the speed will depend on the bandwidth limitations on the client side.

## Limitations of the recognition process

This system has the following recognition limitations

- at present, we have only 90,000 primary verb forms in the verb database, which are commonly found in Sanskrit literature. Though it is very unlikely that ordinary Sanskrit literature will overshoot this list, yet the system is likely to start processing a verb as *subanta* if not found in the database
- the system will wrongly mark a verb with *upasarga* or derived by other derivational process as *subanta* as it will not be found in the verb database. A separate research is underway to comprehensively tag all Sanskrit verbs. The benefits of that research will also help this system in improving performance
- this work assumes initial *sandhi* processing, without which, some results may turn out to be incorrect A separate research is currently being done on *sandhi* splitting on identical

platform. Therefore the next release of the system will include the capability to preprocess for *sandhi* joins as well for better results

- we have 519 *avyayas* listed in our *avyaya* database. It is not enough for *avyaya* recognition in ordinary Sanskrit literature. In this case, the system is likely to start processing an *avyaya* as *subanta*, if it is not found in *avyaya* database.
- some forms ending in primary affixes look like *subanta* but they are originally *avyayas*. For example: पठितुम्, गत्वा, आदाय, विहस्य etc. System will incorrectly recognize and process them as *subanta*.
- many *subantas* (basically *śtr pratyayānta* in locative singular) look like verbs these *subantas* will be wrongly recognized as verbs for example, भवति, गच्छति, पठति, चलति etc.

## Limitations of the analysis process

The system has the following analysis limitations

- Same forms are available in the dual of nominative and accusative cases, for example, (रामौ ), dual of instrumental, dative and ablative cases, for example (रामाभ्याम्) plural of dative and ablative cases, for example ( रामेभ्यः ), dual of genitive and locative cases, for example (रामयोः ). In neuter gender as well, the nominative and accusative singular forms may be identical as in for example पुस्तकम् (1-1 and 2-1). In such cases, the system will give all possible results as in

रामौ	=	औ	[प्र./ द्वि. द्विव.]
रामाभ्याम्	=	भ्याम्	[तृ./च./पं. द्विव.]
रामेभ्यः	=	भ्यस्	[च./पं. बहुव.]
रामयोः	=	ओस्	[ष./स. द्विव.]
पुस्तकम्	=	सु/अम्	[प्र./द्वि. एकव.]
हरेः	=	डसि/डस्	[पं./ष. एकव.]

- some *kṛdanta* forms (generally *lyap*, *tumun*, and *ktvā* suffix ending) look like *subanta* (for example - विहस्य पठित्वा, गत्वा, पठितुम्, गन्तुम्, नेतुम्, प्रदाय, विहाय etc.). In such cases, the system may give wrong results as:

विहस्य = विह + डस् षष्ठी एकवचन  
पठित्वा = पठित्वा + सु प्रथमा एकवचन  
गत्वा = गत्वा + सु प्रथमा एकवचन  
पठितुम् = पठितु + अम् द्वितीया एकवचन  
गन्तुम् = गन्तु + अम् द्वितीया एकवचन  
नेतुम् = नेतु + अम् द्वितीया एकवचन  
प्रदाय = प्रद + डे चतुर्थी एकवचन  
विहाय = विह + डे चतुर्थी एकवचन

To solve this problem, we are trying to store the *kṛdanta* forms of the 500 commonly found verb roots.

- This system does not have gender information for *prātipadikas*, nor does it attempt to guess the gender. A separate research is underway in gender identification and analysis.

### Ambiguity resolution strategies

Currently this system is giving multiple results in ambiguous cases, which will be solved by analyzing the context at the sentence level. A doctoral work to analyze karaka mechanically is nearing its completion. The feedback from the karaka component will be taken to disambiguate between multiple results in this case. In addition, we will be adding some constraints in the forms of additional rules or data to minimize ambiguity.

### Accuracy of results

We have tested on 10 separate files (collected from Sanskrit magazines, Pāñcatantra story and other resources) and did an analysis of the correct and incorrect results as follows -

File	Description	Words	Result	Time
------	-------------	-------	--------	------

			(%)	
Sanskrit corpus-1	Stories from the magazine <i>sambhāṣaṇa sandeśaḥ</i>	560	95	2 sec.
Sanskrit corpus-2	Stories from the magazine <i>sambhāṣaṇa sandeśaḥ</i>	450	90	1.5 sec.
Sanskrit corpus-3	Story collected from books for 10 <sup>th</sup> class CBSCE board	1023	92	3.5 sec.
Sanskrit corpus-4	Story collected from books for 10 <sup>th</sup> class CBSCE board	650	93	2 sec.
Sanskrit corpus-5	Corpus collected from the magazine <i>sambhāṣaṇa sandeśaḥ</i>	478	87	1.4 sec
Sanskrit corpus-6	Corpus collected from the magazine <i>sambhāṣaṇa sandeśaḥ</i>	705	90.5	2.4 sec.
Sanskrit corpus-7	Sanskrit story <i>catura śṛgālaḥ</i> and other from Pañcatantra	823	85	3 sec.
Sanskrit corpus-8	Sanskrit story ' <i>yasya buddhirtasya balam</i> ' and ' <i>deva svāminah</i> ' from	1010	96.75	3.4 sec.

	Pañcatantra			
Sanskrit corpus-9	Sanskrit story ‘trayoh dhūrtāḥ’ from Pañcatantra	840	91.25	3 sec.
Sanskrit corpus-10	Sanskrit story ‘catura śaśakah’ from Pañcatantra	903	96	3 sec.
<b>Average</b>		<b>91.65</b>		

**Table-5.1 : Test Sanskrit corpus**

So the average accuracy of the SRAS at this point is **91.65%** which will further improve with several additions to the rule base, example base and other linguistic resource files.

### Sample-1

The test corpus (corpus-2) with 232 words gave the following result-

चाचा नेहरु:

कस्य अस्ति एतत् चित्रम् ? किम् जानासि ? एषः बालानाम् प्रियः चाचा नेहरुः अस्ति । एषः महापुरुषः स्वतन्त्र-भारतस्य प्रथमः प्रधानमन्त्री अभवत् । अस्य परिधाने रक्तः शतपत्री गुलाबस्य पुष्पम् शोभते । नेहरु-महोदयस्य जन्मः इलाहाबादनगरस्य समृद्ध-परिवारे अभवत् । बाल्यात् तस्य रुचिः स्वाध्याये आसीत् । सः बाल्यावस्थायाम् एव इंग्लैण्डदेशं अगच्छत् । तत्र एव विद्याभ्यासम् च अकरोत् । नेहरु-महोदयस्य पत्नी श्रीमती कमला नेहरु एका साहसशीला विनम्रा च महिला आसीत् । स्वतन्त्रतायाः सर्वेषु कार्येषु तस्याः सहयोगः आसीत् । यदा जलियांवालाबागहत्याकाण्डः अभवत् तदा नेहरु-महोदयस्य गान्धि-महोदयेन सह सम्पर्कः अभवत् । नेहरुः स्वस्वास्थ्य-विषये सदैव जागरूकः आसीत् । सः कर्मवीरः आसीत् । सः वस्तुतः स्वतन्त्रता-सैनिकः आसीत् । कारागारे अपि सः समयस्य सदुपयोगम् अकरोत् । कारागारात् सः स्वपुत्रीं इन्दिरां प्रति अनेकानि पत्राणि अलिखत् । सर्वेषु पत्रेषु भारतस्य इतिहासस्य सभ्यतायाः च विषये चर्चा अस्ति । अद्य अपि



तानि पत्राणि छात्रेभ्यः ज्ञानप्रदानं कुर्वन्ति । कारागारे एव सः स्वाम् आत्मकथाम् अलिखत् । एषा प्रसिद्धा आत्मकथा अस्ति । अद्य छात्राः ताम् पठन्ति गर्वम् च अनुभवन्ति । १५ अगस्त १९४७ तमे वर्षे राष्ट्रम् प्रति प्रथमे सन्देशे सः अकथयत्— अहम् स्वतन्त्र-भारतस्य प्रथमः सेवकः अस्मि ।" संसारे सः भारतस्य प्रतिनिधिः आसीत् । भारतः नेहरूः च पर्यायौ स्तः । तस्य भाषणेषु सर्वदा देशप्रेमस्य धारा अवहत् । बालेषु आसीत् तस्य विशेषः स्नेहः । अनेन कारणेन तस्य जन्मदिवसः अद्य अपि बालदिवसरूपे प्रसिद्धः भवति । विद्यालयेषु बालेभ्यः समारोहाः भवन्ति । नवम्बर-मासस्य १४ तारिकायाः ते प्रतीक्षां कुर्वन्ति । बालाः आगच्छत ! प्रतिज्ञां कुरुत ! भारतस्य यम् स्वप्नम् नेहरू-महोदयः अपश्यम् तम् पूरयत ।

### Output after recognition and analysis of input text

{ चाचा [चाचा+सु प्रथमा एकवचन] नेहरूः [नेहरू+सु प्रथमा एकवचन] कस्य [क+स्य डस्, षष्ठी एकवचन] [अस्ति\_VERB] एतत् [एतद्+सु प्रथमा एकवचन] चित्रम् [चित्र+अम् द्वितीया एकवचन] [?\_PUNCT] किम् [कि+अम् द्वितीया एकवचन] [जानासि\_VERB] [?\_PUNCT] एषः [एष+सु प्रथमा एकवचन] बालानाम् [बाल+आम् षष्ठी बहुवचन] प्रियः [प्रि+अस् प्रथमा बहुवचन] चाचा [चाचा+सु प्रथमा एकवचन] नेहरूः [नेहरू+सु प्रथमा एकवचन] [अस्ति\_VERB] [!\_PUNCT] एषः [एष+सु प्रथमा एकवचन] महापुरुषः [महापुरुष+सु प्रथमा एकवचन] स्वतन्त्रभारतस्य [स्वतन्त्रभारत+स्य डस्, षष्ठी एकवचन] प्रथमः [प्रथम+सु प्रथमा एकवचन] प्रधानमन्त्री [प्रधानमन्त्री+सु प्रथमा एकवचन] [अभवत्\_VERB] [!\_PUNCT] अस्य [अ+स्य डस्, षष्ठी एकवचन] परिधाने [परिधान्+डे चतुर्थी एकवचन] रक्तः [रक्त+सु प्रथमा एकवचन] शतपत्री [शतपत्री+सु प्रथमा एकवचन] गुलाबस्य [गुलाब+स्य डस्, षष्ठी एकवचन] पुष्पम् [पुष्प+अम् द्वितीया एकवचन] [शोभते\_VERB] [!\_PUNCT] नेहरूमहोदयस्य [नेहरूमहोदय+स्य डस्, षष्ठी एकवचन] जन्मः [जन्म+सु प्रथमा एकवचन] इलाहाबादनगरस्य [इलाहाबादनगर+स्य डस्, षष्ठी एकवचन] समृद्धपरिवारे [समृद्धपरिवार+ङि सप्तमी एकवचन] [अभवत्\_VERB] [!\_PUNCT] बाल्यात् [बाल्य+ङसि पञ्चमी एकवचन] तस्य [त+स्य डस्, षष्ठी एकवचन] रुचिः [रुचि+सु प्रथमा एकवचन] स्वाध्याये [स्वाध्यै+डे चतुर्थी एकवचन] [आसीत्\_VERB] [!\_PUNCT] सः [स+सु प्रथमा एकवचन] बाल्यावस्थायाम् [बाल्यावस्था+ङि सप्तमी एकवचन] [एव\_AV] इंग्लैण्डदेशं [इंग्लैण्डदेश+अम् द्वितीया एकवचन] [अगच्छत्\_VERB] [!\_PUNCT] [तत्र\_AV] [एव\_AV] विद्याभ्यासम् [विद्याभ्यास+अम् द्वितीया एकवचन] [च\_AV]

[अकरोत्-VERB] [I-PUNCT] नेहरूमहोदयस्य [नेहरूमहोदय+स्य डस्, षष्ठी एकवचन] पत्नी [पत्नी+सु प्रथमा एकवचन] श्रीमती [श्रीमती+सु प्रथमा एकवचन] कमला [कमला+सु प्रथमा एकवचन] नेहरू [नेहरू+औ/औट् प्रथमा द्वितीया द्विवचन] एका [एका+सु प्रथमा एकवचन] साहसशीला [साहसशीला+सु प्रथमा एकवचन] विनम्रा [विनम्रा+सु प्रथमा एकवचन] [च-AV] महिला [महिला+सु प्रथमा एकवचन]

[आसीत्-VERB] [I-PUNCT] स्वतन्त्रतायाः [स्वतन्त्रता+डसि/डस् पञ्चमी/षष्ठी एकवचन] सर्वेषु [सर्व+सुप् सप्तमी बहुवचन] कार्येषु [कार्य+सुप् सप्तमी बहुवचन] तस्याः [तस्+डसि पञ्चमी/षष्ठी एकवचन] सहयोगः [सहयोग+सु प्रथमा एकवचन] [आसीत्-VERB] [I-PUNCT] [यदा-AV] जलियांवालाबागहत्याकाण्डः [जलियांवालाबागहत्याकाण्ड+सु प्रथमा एकवचन] [अभवत्-VERB] [तदा-AV] नेहरूमहोदयस्य [नेहरूमहोदय+स्य डस्, षष्ठी एकवचन] गान्धिमहोदयेन [गान्धिमहोदय+टा तृतीया एकवचन] [सह-AV] सम्पर्कः [सम्पर्क+सु प्रथमा एकवचन] [अभवत्-VERB] [I-PUNCT] नेहरूः [नेहरू+शस् द्वितीया बहुवचन] स्वस्वास्थ्यविषये [स्वस्वास्थ्यविषि+डे चतुर्थी एकवचन] [सदैव-AV] जागरूकः [जागरूक+सु प्रथमा एकवचन] [आसीत्-VERB] [I-PUNCT] सः [स+सु प्रथमा एकवचन] कर्मवीरः [कर्मवीर+सु प्रथमा एकवचन] [आसीत्-VERB] [I-PUNCT] सः [स+सु प्रथमा एकवचन] [वस्तुतः-AV] स्वतन्त्रतासैनिकः [स्वतन्त्रतासैनिक+सु प्रथमा एकवचन] [आसीत्-VERB] [I-PUNCT] कारागारे [कारागार+ङि सप्तमी एकवचन] [अपि-AV] सः [स+सु प्रथमा एकवचन] समयस्य [समय+स्य डस्, षष्ठी एकवचन] सदुपयोगम् [सदुपयोग+अम् द्वितीया एकवचन] [अकरोत्-VERB] [I-PUNCT] कारागारात् [कारागार+डसि पञ्चमी एकवचन] सः [स+सु प्रथमा एकवचन] स्वपुत्रीं [स्वपुत्री+अम् द्वितीया एकवचन] इन्दिरां [इन्दिरा+अम् द्वितीया एकवचन] [प्रति-AV] अनेकानि [अनेक+जस् प्रथमा द्वितीया बहुवचन] पत्राणि [पत्र+जस् प्रथमा द्वितीया बहुवचन] [अलिखत्-VERB] [I-PUNCT] सर्वेषु [सर्व+सुप् सप्तमी बहुवचन] पत्रेषु [पत्र+सुप् सप्तमी बहुवचन] भारतस्य [भारत+स्य डस्, षष्ठी एकवचन] इतिहासस्य [इतिहास+स्य डस्, षष्ठी एकवचन] सभ्यतायाः [सभ्यता+डसि/डस् पञ्चमी/षष्ठी एकवचन] [च-AV] विषये [विषि+डे चतुर्थी एकवचन] चर्चा [चर्चा+सु प्रथमा एकवचन] [अस्ति-VERB] [I-PUNCT] [अद्य-AV] [अपि-AV] तानि [त+जस् प्रथमा द्वितीया बहुवचन] पत्राणि [पत्र+जस् प्रथमा द्वितीया बहुवचन] छात्रेभ्यः [छात्र+भ्यस् चतुर्थी पञ्चमी बहुवचन] ज्ञानप्रदानं [ज्ञानप्रदन्+अम् द्वितीया एकवचन] [कुर्वन्ति-VERB] [I-PUNCT] कारागारे [कारागार+ङि सप्तमी एकवचन] [एव-AV] सः [स+सु प्रथमा एकवचन] स्वाम् [स्व+ङि सप्तमी एकवचन]

आत्मकथाम् [आत्मकथा+अम् द्वितीया एकवचन] [अलिखत्\_VERB] [I\_PUNCT] एषा [एषा+सु प्रथमा एकवचन] प्रसिद्धा [प्रसिद्ध+टा तृतीया एकवचन] आत्मकथा [आत्मकथा+सु प्रथमा एकवचन] [अस्ति\_VERB] [I\_PUNCT] [अद्य\_AV] छात्राः [छात्र+जस् प्रथमा बहुवचन] ताम् [ता+अम् द्वितीया एकवचन] [पठन्ति\_VERB] गर्वम् [गर्+अम् द्वितीया एकवचन] [च\_AV] [अनुभवन्ति\_VERB] [I\_PUNCT] अगस्त\_SUBANTA तमे [तम+ङि सप्तमी एकवचन] वर्षे [वर्ष+ङि सप्तमी एकवचन] राष्ट्रम् [राष्ट्र+अम् द्वितीया एकवचन] [प्रति\_AV] प्रथमे [प्रथम+ङि सप्तमी एकवचन] सन्देशे [सन्देश+ङि सप्तमी एकवचन] सः [स+सु प्रथमा एकवचन] [अकथयत्\_VERB] अहम् [अह+अम् द्वितीया एकवचन] स्वतन्त्रभारतस्य [स्वतन्त्रभारत+स्य डस्, षष्ठी एकवचन] प्रथमः [प्रथम+सु प्रथमा एकवचन] सेवकः [सेवक+सु प्रथमा एकवचन] [अस्मि\_VERB] "\_PUNCT ["\_PUNCT] संसारे [संसार+ङि सप्तमी एकवचन] सः [स+सु प्रथमा एकवचन] भारतस्य [भारत+स्य डस्, षष्ठी एकवचन] प्रतिनिधिः [प्रतिनिधि+सु प्रथमा एकवचन] [आसीत्\_VERB] [I\_PUNCT] भारतः [भारत+सु प्रथमा एकवचन] नेहरूः [नेहरू+शस् द्वितीया बहुवचन] [च\_AV] पर्यायौ [पर्यै+औ/औट् प्रथमा द्वितीया द्विवचन] [स्तः\_VERB] [I\_PUNCT] तस्य [त+स्य डस्, षष्ठी एकवचन] भाषणेषु [भाषण+सुप् सप्तमी बहुवचन] [सर्वदा\_AV] देशप्रेमस्य [देशप्रेम+स्य डस्, षष्ठी एकवचन] धारा [धारा+सु प्रथमा एकवचन] [अवहत्\_VERB] [I\_PUNCT] बालेषु [बाल+सुप् सप्तमी बहुवचन] [आसीत्\_VERB] तस्य [त+स्य डस्, षष्ठी एकवचन] विशेषः [विशेष+सु प्रथमा एकवचन] स्नेहः [स्नेह+जस्/शस्/ङसि/डस्,प्रथमा/द्वितीया बहुवचन./पञ्चमी./षष्ठी एकवचन] [I\_PUNCT] अनेन [अन+टा तृतीया एकवचन] कारणेन [कारण+टा तृतीया एकवचन] तस्य [त+स्य डस्, षष्ठी एकवचन] जन्मदिवसः [जन्मदिवस+सु प्रथमा एकवचन] [अद्य\_AV] [अपि\_AV] बालदिवसरूपे [बालदिवसरूप+ङि सप्तमी एकवचन] प्रसिद्धः [प्रसिद्ध+जस्/शस्/ङसि/डस्,प्रथमा/द्वितीया बहुवचन./पञ्चमी./षष्ठी एकवचन] [भवति\_VERB] [I\_PUNCT] विद्यालयेषु [विद्यालय+सुप् सप्तमी बहुवचन] बालेभ्यः [बाल+भ्यस् चतुर्थी पञ्चमी बहुवचन] समारोहाः [समारोह+जस्/शस्,प्रथमा/द्वितीया बहुवचन] [भवन्ति\_VERB] [I\_PUNCT] नवम्बरमासस्य [नवम्बरमास+स्य डस्, षष्ठी एकवचन] तारिकायाः [तारिका+ङसि/डस् पञ्चमी/षष्ठी एकवचन] [ते\_AV] प्रतीक्षां [प्रतीक्स्+आम् षष्ठी बहुवचन] [कुर्वन्ति\_VERB] [I\_PUNCT] बालाः [बाल+जस् प्रथमा बहुवचन] [आगच्छत\_VERB] [I\_PUNCT] प्रतिज्ञां [प्रतिजम्+आम् षष्ठी बहुवचन] [कुरुत\_VERB] [I\_PUNCT] भारतस्य [भारत+स्य डस्, षष्ठी एकवचन] यम् [य+अम् द्वितीया एकवचन] स्वप्नम् [स्वप्न+अम्

द्वितीया एकवचन] नेहरूमहोदयः [नेहरूमहोदि+अस् प्रथमा बहुवचन] [अपश्यम्\_VERB] तम् [त+अम् द्वितीया एकवचन] [पूरयत\_VERB] [।\_PUNCT] }

## Processing speed

System took 2 secs for processing 232 words on an acer machine with Intel Pentium 4, CPU 1.80GHz, 256 MB of RAM.

## Future Research and Development

The SRAS has tremendous potentials in the field of Sanskrit NLP and M(A)TS. Some of the immediate and future applications of the system are discussed below –

## Machine Translation System (MTS)

Major goal of this and other R&D currently in progress is to design an M(A)TS from Sanskrit to Indian languages, which can not be achieved without reverse analysis of subanta padas.

## Self-reading and understanding

The subanta analyzer can alone be used for simplification of Sanskrit texts for simple reading and comprehension.

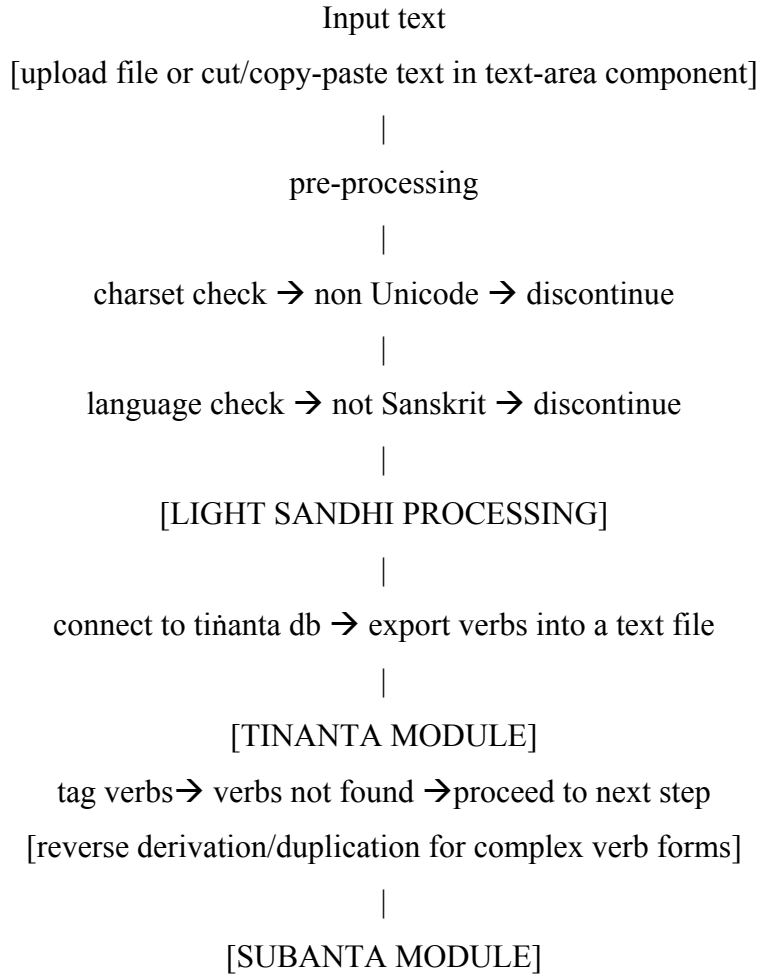
## Sanskrit processing for any purpose

Any further processing of Sanskrit can be done only after *subantas* have been analyzed – for example - *kṛdanta* recognition and analysis, *taddhita* recognition and analysis, *samāsa* recognition and analysis, POS tagging, gender recognition and analysis. Sanskrit sentence discourse analysis can be future areas of R & D. A computational lexicography work on etymological, exegetical methods adopted in *nirukta* of Yāska is another fascinating area to work on. Developing a Sanskrit text search engine like *Aṣṭādhyāyī* , *Mahābhārata* , *Rāmāyana* ,

*purāṇa* , Vedic index and search etc. are other future research and development areas in this field which can be undertaken at the Ph.D. level.

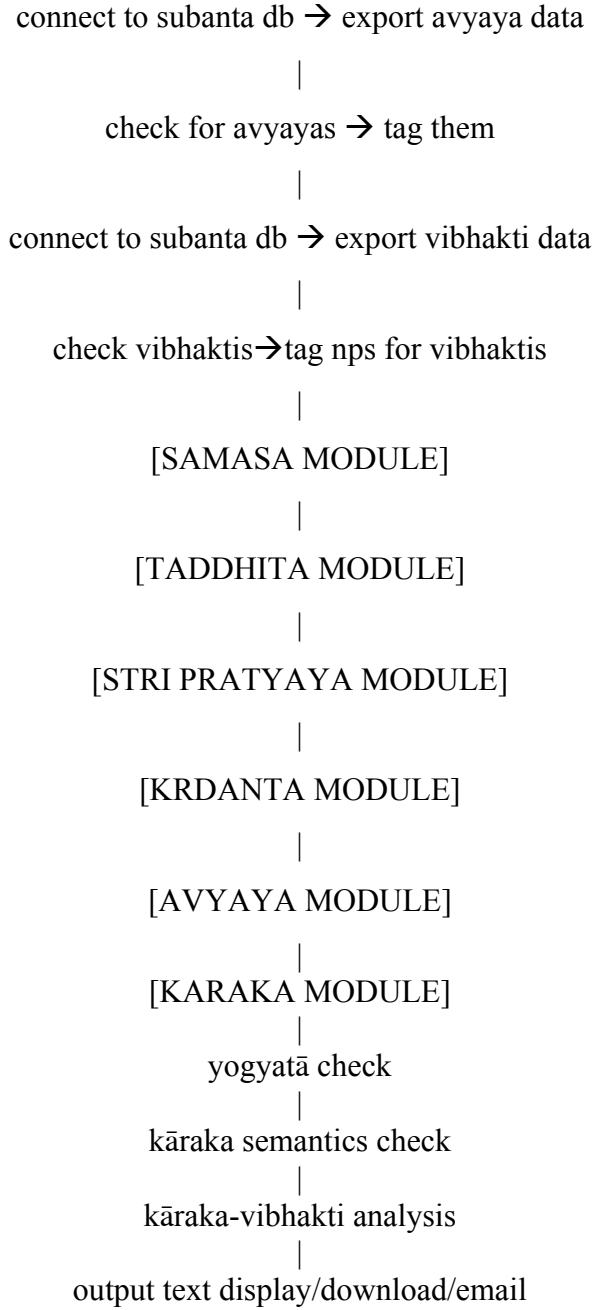
## **Towards developing a Sanskrit Analysis System**

The SRAS can be the first step towards developing a comprehensive analysis system for Sanskrit. Jha<sup>237</sup> and all (2006) presented a Sanskrit Analysis System recently as follows –



---

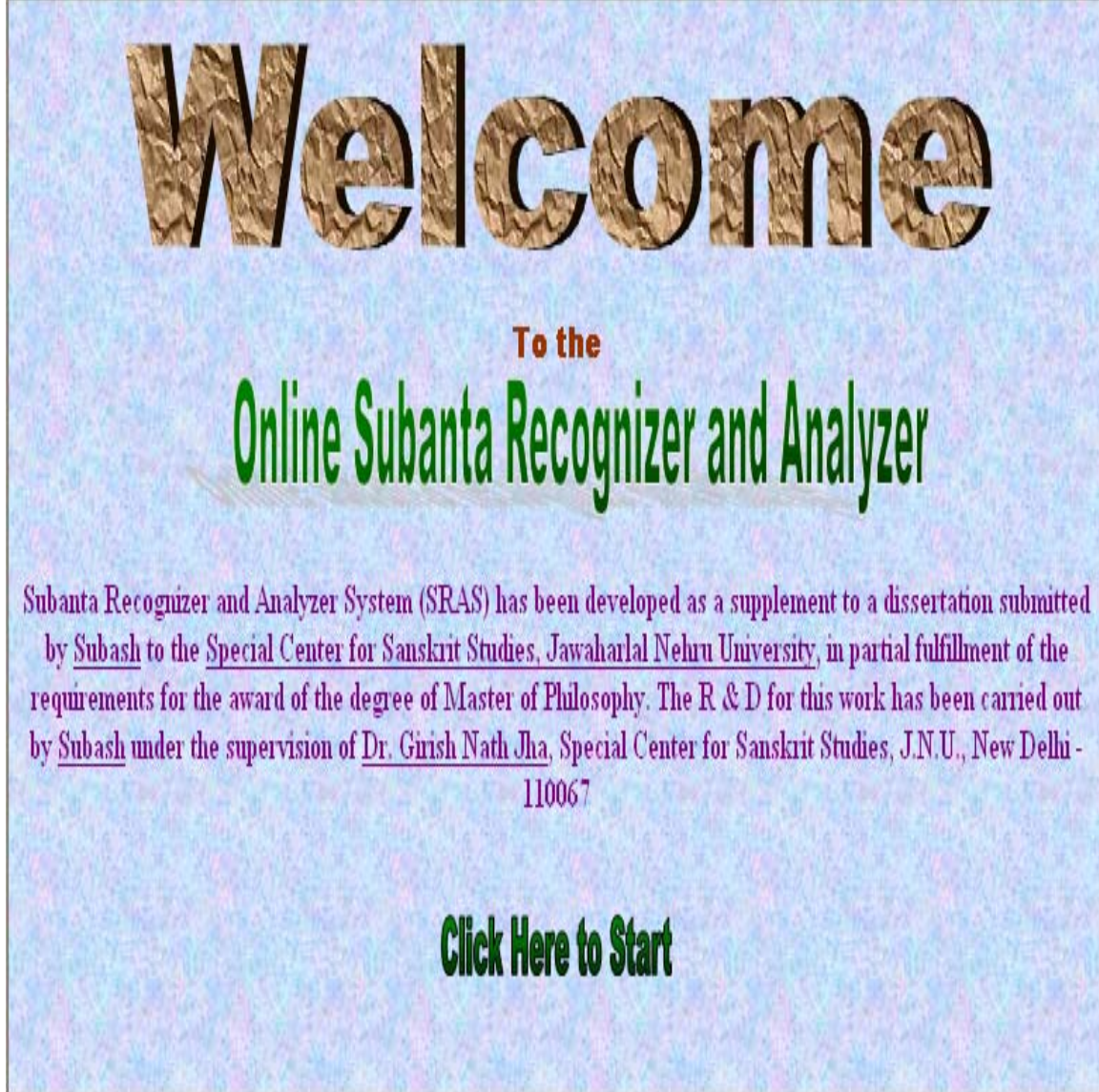
<sup>237</sup> **Towards a Computational analysis system for Sanskrit**” in the proceeding of first National symposium on Modeling and Shallow parsing of Indian Languages at Indian Institute of Technology Bombay pp 25-34 on 2<sup>nd</sup> to 4<sup>th</sup> April 2006



## **APPENDICES**

## APPENDIX- I

### CD Program: Welcome page



## APPENDIX- II



## CD: Home page

[INSTALLATION INSTRUCTIONS](#)[HOW TO RUN SRAS](#)[SEND FEEDBACK](#)

### Installation instructions

You will need to install the following applications for using the SRAS web application -

- Java development environment and runtime
- Apache Tomcat 4.0 as webserver
- Baraha IME (or any other unicode IME) for Sanskrit input

#### Java Environment

Your system must have java installed to support the Java Web Server. If your system does not have JAVA Please [CLICK HERE](#) to install JAVA in your system

#### Apache Tomcat as the Web-Server

Your system must have Apache-Tomcat version 4.0 or above as the web server. If you do not have this, please [CLICK HERE](#) to install Apache Tomcat 4.0 in your system.

#### [Baraha](#) IME (or any other unicode Devanagari IME) for Sanskrit input

You will need to install Baraha software for UTF-8 Sanskrit input. If you do not have this on your computer, please [CLICK HERE](#) to install Baraha in your system.

[INSTALLATION INSTRUCTIONS](#)[HOW TO RUN SRAS](#)[SEND FEEDBACK](#)

## APPENDIX- III

### CD: Contact us page



[INSTALLATION INSTRUCTIONS](#)    [HOW TO RUN SRAS](#)    [SEND FEEDBACK](#)

---

### Subanta Recognizer and Analyzer for Sanskrit (SRAS)

If you have Java and Tomcat installed on your machine, then do the following -

1. Copy the **Subanta** folder from this CD to the "webapps" folder under tomcat installation folder, for example, **C:\Program Files\Apache Tomcat 4.0\webapps**
2. Now start tomcat by clicking on the **Start Tomcat** icon in **C:\Program Files\Apache Tomcat 4.0\bin**  
Or  
by clicking **Start menu>All Programs>Apache Tomcat 4.0>Start Tomcat**.
3. Now, click **run SRAS** to run the application.  
If it does not start automatically, then copy the link **http://localhost:8080/subanta/rsubanta.jsp** to your web browser URL
5. Now, start Baraha IME (by clicking **Start menu>All Programs>Baraha>Baraha Direct**) and set the language to Sanskrit and output format to unicode. You can type using phonetic layout, the instructions for which can be found in the dissertation page4  
Alternatively, you can click the following links to get sample Sanskrit text-
 

[SANSKRIT CORPUS-1](#)    [SANSKRIT CORPUS-2](#)    [SANSKRIT CORPUS-3](#)  
[SANSKRIT CORPUS-4](#)    [SANSKRIT CORPUS-5](#)    [SANSKRIT CORPUS-6](#)  
[SANSKRIT CORPUS-7](#)    [SANSKRIT CORPUS-8](#)    [SANSKRIT CORPUS-9](#)  
[SANSKRIT CORPUS-10](#)

#### Input Instructions:

Please make sure that the input follows the following requirements-

- input must be Devanagari UTF-8 Sanskrit text
- input must be sandhi free
- input can be a word/sentence/complete texts

#### Result analysis

System prints result as output in three colors - Black, Blue and Red

- analyzed subanta
- non subanta (punctuations/avyaya/verb)
- un-analyzed subanta

The result has the following structure

- subanta pada [analysis] in Black color
- [non-subanta with tag ] in Blue color
- unanalyzed subanta with ' SUBANTA' tag in Red color

For details, see the chapter-II titled *Subanta processing of Panini*  
For analysis of Subanta-padas, see the chapter-III titled *Subanta analysis*

[INSTALLATION INSTRUCTIONS](#)    [HOW TO RUN SRAS](#)    [SEND FEEDBACK](#)

## APPENDIX- IV

### Webpage user interface

## सङ्गणक द्वारा सुबन्त पहचान और प्रकृति-प्रत्यय विभाग

यह कैसे कार्य करता है [देवनागरी लेखन की सहायता के लिये सिस्टम की सीमायें सम्पर्क करें](#)

सुबन्त पहचान एवं प्रकृति-प्रत्यय विभाग के लिये कृपया संस्कृत वाक्य, गद्य या संस्कृत पद लिखें

प्रकृति-प्रत्यय विभाग के लिये यहाँ क्लिक करें

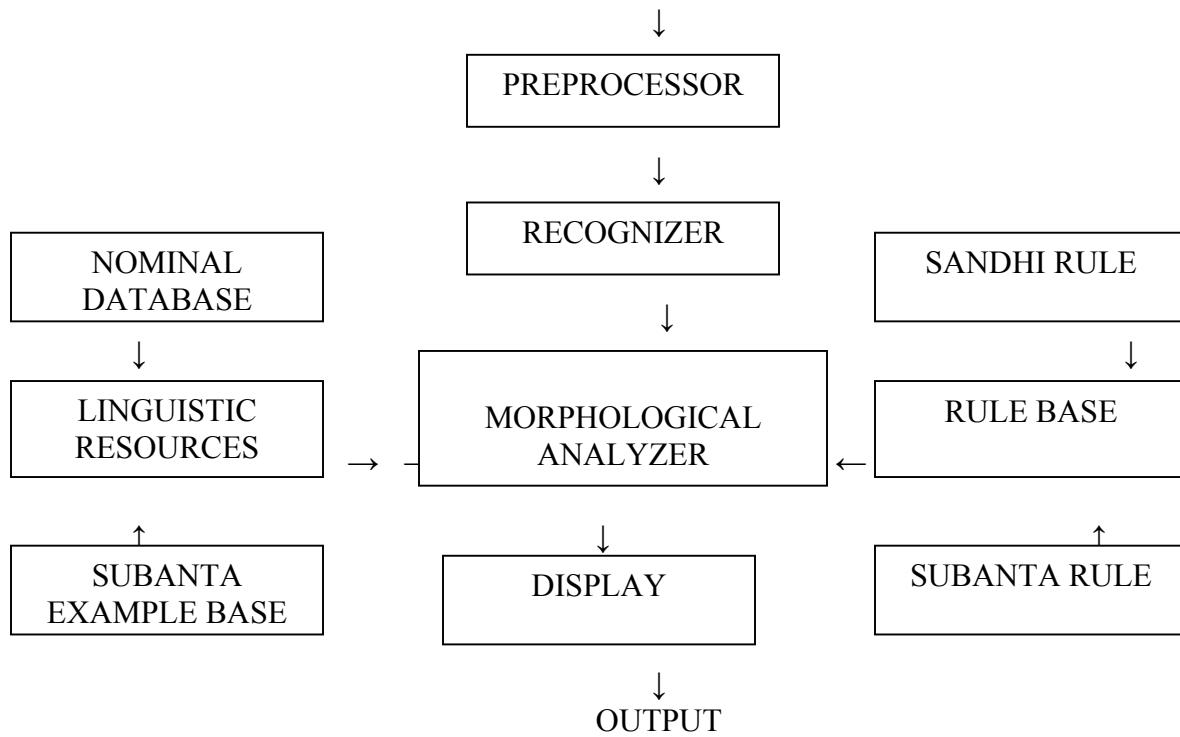
Run in debug mode ☐

### परिणाम

## APPENDIX- V

### The SRAS modules

SANSKRIT TEXT (UTF-8)





## APPENDIX- VI: Test sample-1

http://localhost:8080/subanta/rsubanta.jsp - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Reload Home Search Favorites SnagIt

Search the Web Search Address http://localhost:8080/subanta/rsut Links

निमिषमात्रं विचिन्त्य भाषितवान् - " पुण्यसलिला भागीरथी भवतः जटासम्भारात् निर्गत्य पाटलीपुत्रपर्यन्तं तीरवासिनां पापानि नाशयति । पश्चात् सा वामतः वलति । अनन्तरं पूर्वसागरेण सङ्गच्छते . एवं अधनाशिन्याः गङ्गायाः पावनतीर्थलाभात् वञ्चिताः । अतः अहं प्रार्थये - 'गङ्गामातुः पादतीर्थं दक्षिणभारते अपि उपलब्धं भवेत्' इति । तेन तीर्थेन वयं सर्वे दक्षिणभारतीयाः अपि पापविमुक्ताः भवेम " इति। शङ्करः भक्तप्रियः । " तथा अस्तु " इति हस्तम् उद्धृत्य सः अवदत् ।

हर्षोत्फुल्लः सुकेशी अभाषत - " गङ्गायाः प्रवाहमार्गः परिवर्तितः भवेत्, समग्रे दक्षिणभारते सः प्रसृतः भवेत् च इति अपेक्षे

प्रकृति-प्रत्यय विभाग के लिये यहाँ क्लिक करें Run in debug mode ☐

### परिणाम

{ सम्भाषण\_SUBANTA सन्देश\_SUBANTA [पुरा\_AV] राजा [राजन्+टा तृतीया एकवचन] सगरेण [सगर+टा तृतीया एकवचन] अश्रवमेधयजः [अश्रवमेधयज+सु प्रथमा एकवचन] [अनुतिष्ठतः VERB] [I\_PUNCT] यज्ञीयः [यज्ञीय+सु प्रथमा एकवचन] अश्रवः [अश्रू+अस् प्रथमा बहुवचन] कपिलमुनेः [कपिलमुनि+इति पञ्चमी/षष्ठी एकवचन] आश्रमे [आश्रम+इति सप्तमी एकवचन] निबध्दः [निबध्द+सु प्रथमा एकवचन] [I\_PUNCT] सगरस्य [सगर+स्य डस्, षष्ठी एकवचन] सहस्रं [सहस्र+अम् द्वितीया एकवचन] पुत्राः [पुत्र+जस् प्रथमा बहुवचन] पृथ्वीम् [पृथ्वी+अम् द्वितीया एकवचन] उदखनन् [उदखनम्+सु प्रथमा एकवचन] [I\_PUNCT] अश्रवं [अश्रव+अम् प्रथमा एकवचन] [च\_AV] प्राप्नुवन् [प्राप्नुवम्+सु प्रथमा एकवचन] [I\_PUNCT] तैः [तद्+भिः तृतीया बहुवचन] सगरपुत्रैः [सगरपुत्र+भिस् तृतीया बहुवचन] कपिलमुनौ [कपिलमुनि+इति सप्तमी एकवचन] चौर्यकर्म\_SUBANTA आरोपितम् [आरोपित+अम् द्वितीया एकवचन] [I\_PUNCT] क्रुध्दः [क्रुध्द+सु प्रथमा एकवचन] कपिलः [कपिल+सु प्रथमा एकवचन] सगरपुत्रान् [सगरपुत्र+शस् द्वितीया बहुवचन] शपन् [शपम्+सु प्रथमा एकवचन] तान् [तद्+शस् द्वितीया बहुवचन] भस्मसात् [भस्मस+इति पञ्चमी एकवचन] [अकरोत् VERB] [I\_PUNCT] पश्चात् [पश्च+इति पञ्चमी एकवचन] सगरवंशीयः [सगरवंशीय+सु प्रथमा एकवचन] भागीरथः [भागीरथ+सु प्रथमा एकवचन] महता [महता+सु प्रथमा एकवचन] तपोबलेन [तपोबल+टा तृतीया एकवचन] गङ्गा [गङ्गा+सु प्रथमा एकवचन] स्वर्गात् [स्वर्ग+इति पञ्चमी एकवचन] भूलोकम् [भूलोक+अम् द्वितीया एकवचन] [आनयत् VERB] [I\_PUNCT] शापग्रस्तानां [शापग्रस्त+आम् षष्ठी बहुवचन] भस्मीभूतानां

Local intranet

start classes Start Tomcat 2 Internet Explo... 4 Notepad 2:35 PM

## APPENDIX- VII

### Test sample-2

http://localhost:8080/subanta/rsubanta.jsp - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Refresh Mail Print Word Pad Address Bar Links SnagIt

Search the Web Search Address http://localhost:8080/subanta/rsut Links SnagIt

### परिणाम

{ समाजः [समाह्+जस्/शस्/डसि/डस्/प्रथमा/द्वितीया बहुवचन/पञ्चमी/षष्ठी एकवचन] अशोकस्य [अशोक+स्य डस्, षष्ठी एकवचन] जन्मदिनम् [जन्मदिन+अम् द्वितीया एकवचन] आचार्यमाणम् [आचार्यमन्+अम् द्वितीया एकवचन] [आसीत् VERB] [I\_PUNCT] [तत्र AV] अधीनाः [अधीन+जस् प्रथमा बहुवचन] नृपाः [नृप+जस् प्रथमा बहुवचन] भागं [भाग+अम् द्वितीया एकवचन] गृहीतवन्तः [गृहीतवन्त+सु प्रथमा एकवचन] आसन् [आसम्+सु प्रथमा एकवचन] [I\_PUNCT] [तदा AV] राजा [राजन+सु प्रथमा एकवचन] अशोकः [अशोक+सु प्रथमा एकवचन] एकाम् [एक + आम् षष्ठी एकवचन] घोषणां [घोष+आम् षष्ठी बहुवचन] कारितवान् [कारितव+शस् द्वितीया बहुवचन] अत्युत्तमस्य [अत्युत्तम+स्य डस्, षष्ठी एकवचन] राज्ञः [राज्ञ+शस्/डसि/डस्/द्वितीया बहुवचन/पञ्चमी/षष्ठी एकवचन] सम्माननं [सम्मानन+अम् द्वितीया एकवचन] करिष्यते [करिष्यत+डि सप्तमी एकवचन] [I\_PUNCT] [इति AV] [I\_PUNCT] [इति AV] एकैकम् [एकैक+अम् द्वितीया एकवचन] [अपि AV] नृपम् [नृप+अम् द्वितीया एकवचन] आहुय SUBANTA तत्तद्राज्यस्य [तत्तद्राज्य+स्य डस्, षष्ठी एकवचन] विषयम् [विषय+अम् द्वितीया एकवचन] अपृच्छत् [अपृच्छ+सु प्रथमा एकवचन] [I\_PUNCT] [तदा AV] नृपह् SUBANTA स्वीयस्य [स्वीय+स्य डस्, षष्ठी एकवचन] राज्यस्य [राज्य+स्य डस्, षष्ठी एकवचन] प्रगतिम् [प्रगति+अम् द्वितीया एकवचन] उत्साहेन [उत्साह+टा तृतीया एकवचन] वर्णितवन्तः [वर्णितवन्त+सु प्रथमा एकवचन] [I\_PUNCT] उत्तराज्यस्य [उत्तराज्य+स्य डस्, षष्ठी एकवचन] राजा [राजन+सु प्रथमा एकवचन] गर्वणं [गर्व+टा तृतीया एकवचन] [अवदत् VERB] प्रभो SUBANTA अस्मिन् [इदम्+डि सप्तमी एकवचन] वर्षे [वर्ष+डि सप्तमी एकवचन] मम SUBANTA राज्ये [राज्+डि चतुर्थी एकवचन] वृष्टिः [वृष्टि+सु प्रथमा एकवचन] [I\_PUNCT] सस्वसमृद्धिश्च SUBANTA सम्यक् [सम्यह्+सु प्रथमा एकवचन] [अस्ति VERB] [I\_PUNCT] [अतः AV] जनेभ्यः [जन+भ्यस् चतुर्थी पञ्चमी बहुवचन] अत्यधिकं [अत्यधिक+अम् द्वितीया एकवचन] करशुक्लं [करशुक्ल+अम् द्वितीया एकवचन] स्वीकर्तुं [स्वीकर्तु+अम् द्वितीया एकवचन] शक्यते [शक्यत+डि सप्तमी एकवचन] [I\_PUNCT] एतस्मात् [एतद्+डसि पञ्चमी एकवचन] कारणात् [कारण+डसि पञ्चमी एकवचन] मम SUBANTA राज्यस्य [राज्य+स्य डस्, षष्ठी एकवचन] आयः [आ+अस् प्रथमा बहुवचन] त्रिगुणितः [त्रिगुणित+सु प्रथमा एकवचन] [अविष्यति VERB] [I\_PUNCT] [इति AV] [I\_PUNCT] [अनन्तरं AV] दक्षिणराज्यस्य [दक्षिणराज्य+स्य डस्, षष्ठी एकवचन] नृपः [नृप+सु प्रथमा एकवचन] [I\_PUNCT] मम SUBANTA राज्ये [राज्+डि चतुर्थी एकवचन] अहं [अस्मद्+सु प्रथमा एकवचन] कशुल्कस्य [कशुल्क+स्य डस्, षष्ठी एकवचन] संग्रहणं [संग्रहण+अम् द्वितीया एकवचन] सुवर्नरूपेण [सुवर्नरूप+टा तृतीया एकवचन] कृतवान् [कृतव+शस् द्वितीया बहुवचन] [I\_PUNCT] अस्मिन् [इदम्+डि सप्तमी एकवचन] वर्षे [वर्ष+डि सप्तमी एकवचन] मम SUBANTA राज्ये [राज्+डि चतुर्थी एकवचन] भाण्डरे [भाण्डर+डि सप्तमी एकवचन] द्विगुणितं [द्विगुणित+अम् द्वितीया एकवचन] सुवर्णं [सुवर्ण+अम् द्वितीया एकवचन] सङ्गृहीतम्

Done Local intranet

start 2 Window... Start Tomcat 2 Internet... 4 Notepad APPENDICE... 2:38 PM



## APPENDIX-VIII: Test sample -3

http://localhost:8080/subanta/rsubanta.jsp - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites Mail Print Word Pad Internet Options

Search the Web Search Address http://localhost:8080/subanta/rsut Links SnagIt

### परिणाम

{ प्रवाचकः [प्रवाचक+सु प्रथमा एकवचन] [\_PUNCT] संस्कृतविभागः [संस्कृतविभाग+सु प्रथमा एकवचन] काशीहिन्दूविश्वविद्यालयः [काशीहिन्दूविश्वविद्यालि+अस् प्रथमा बहुवचन] [\_PUNCT] वाराणसी [वाराणसी+सु प्रथमा एकवचन] [किं\_AV] बहिरैव SUBANTA स्थास्वसि SUBANTA [?] PUNCT [अन्तः\_AV] आगच्छा [आगच्छा+सु प्रथमा एकवचन] [\_PUNCT] अद्याश्च SUBANTA " PUNCT [ PUNCT] वसुधायाः [वसुधा+इसि/इस् पञ्चमी/षष्ठी एकवचन] उक्त्या [उक्त्या+ता तृतीया एकवचन] अवधानं [अवध+शस् द्वितीया बहुवचन] [मे\_AV] भग्नम् [भग्न+अम् द्वितीया एकवचन] [I\_PUNCT] अतिथिप्रकोष्ठे [अतिथिप्रकोष्ठ+इ सप्तमी एकवचन] माम् SUBANTA उपवेश्य SUBANTA एषा SUBANTA SUBANTA अहम् [अस्मद्+सु प्रथमा एकवचन] आगच्छामि [आगच्छामि+इ सप्तमी एकवचन] [ PUNCT] इदत् [इद+सु प्रथमा एकवचन] उक्त्या [उक्त्या+ता तृतीया एकवचन] [अन्तः\_AV] प्राविशत् [प्राविशत्+सु प्रथमा एकवचन] वसुधा [वसुध+ता तृतीया एकवचन] [I\_PUNCT] गते [गत+इ सप्तमी एकवचन] सोमवासरे [सोमवासर+इ सप्तमी एकवचन] रेलस्थानके [रेलस्थानक+इ सप्तमी एकवचन] अस्मात् [अस्मद्+इसि पञ्चमी एकवचन] वसुधायाः [वसुधा+इसि/इस् पञ्चमी/षष्ठी एकवचन] मेलनं [मेलन+अम् द्वितीया एकवचन] जातम् [जात+अम् द्वितीया एकवचन] [आसीत्\_VERB] [I\_PUNCT] यस्माद् [यस्मद्+इसि पञ्चमी एकवचन] रेलयानाद् [रेलयान+इसि पञ्चमी एकवचन] अहम् [अस्मद्+सु प्रथमा एकवचन] अवतरामि [अवतराम्+इ सप्तमी एकवचन] [स्म\_AV] तेनैव SUBANTA गच्छन्तं [गच्छन्त+अम् द्वितीया एकवचन] पुत्रं SUBANTA आप्नुं [आप्नु+अम् द्वितीया एकवचन] [तत्र\_AV] उपस्थिता [उपस्थिता+सु प्रथमा एकवचन] [आसीत्\_VERB] सा [तद्+सु प्रथमा एकवचन] [I\_PUNCT] विशतिवर्षेभ्यः [विशतिवर्षेभ्यस् चतुर्थी पञ्चमी बहुवचन] [प्राक्\_AV] समामे [समाम+इ सप्तमी एकवचन] वर्गे [वर्गे+इ सप्तमी एकवचन] उपविश्य SUBANTA आवं [आ+अम् द्वितीया एकवचन] सहाध्ययनं [सहाध्ययन+अम् द्वितीया एकवचन] कुर्वन्तीः [कुर्वन्ती+सु प्रथमा एकवचन] आस्व SUBANTA दर्शनमात्रेण [दर्शनमात्र+ता तृतीया एकवचन] [एव\_AV] आगभ्यां [अस्मद्+भ्याम् तृतीया, चतुर्थी, पञ्चमी द्विवचन] परस्परपरिचयं [परस्परपरिचय+अम् द्वितीया एकवचन] प्राप्तः [प्राप्त+सु प्रथमा एकवचन] [I\_PUNCT] सा [तद्+सु प्रथमा एकवचन] स्वपुत्रस्य [स्वपुत्र+स्य इस्, षष्ठी एकवचन] [ PUNCT] आलोकस्य [आलोक+स्य इस्, षष्ठी एकवचन] परिचयं [परिचय+अम् द्वितीया एकवचन] कारितवती [कारितवती+सु प्रथमा एकवचन] [I\_PUNCT] आलोकः [आलोक+सु प्रथमा एकवचन] ग्वालियरनगरे [ग्वालियरनगर+इ सप्तमी एकवचन] चिकित्साशास्त्रम् [चिकित्साशास्त्र+अम् द्वितीया एकवचन] अधिते [अधित+इ सप्तमी एकवचन] [I\_PUNCT] रेलयानं [रेलयान+अम् द्वितीया एकवचन] प्रस्थितम् [प्रस्थित+अम् द्वितीया एकवचन] [I\_PUNCT] वसुधा [वसुध+ता तृतीया एकवचन] दृष्टाः [दृष्ट+जस् प्रथमा बहुवचन] तस्याः [तद्+इसि/इस् पञ्चमी, षष्ठी एकवचन] वैयक्तिकजीवनेविषये [वैयक्तिकजीवनेविषये चतुर्थी एकवचन]

Done Local intranet

start 2 Window... Start Tomcat 2 Internet... 4 Notepad APPENDICE... 2:39 PM

## **BIBLIOGRAPHY**



1. Abbi, Anvita. Semantic universals in Indian language, Indian Institute of Advance Study, Shimla, 1994
2. Academy of Sanskrit Research, (ASR) Melkote,
3. Ananthanarayana, H. S., 1970, 'The kāraka theory and case grammar', Indian Linguistics, Poona, 31: 14-27.
4. Anderson, An Introduction to natural languages processing, Pentice Hall India
5. Anderson, John M., 1971, 'the grammar of case: towards a localistic theory', Cambridge University Press, Cambridge.
6. Anusaaraka, [http://www.iiit.net/ltrc/Anusaaraka/anu\\_home.html](http://www.iiit.net/ltrc/Anusaaraka/anu_home.html) (accessed: 27 April 2005).
7. Baraha, Software, <http://www.baraha.com/BarahaIME.htm> (accessed: 12 July 2006).
8. Bhandarkar, R.G., 1924, 'First book of Sanskrit' Radhabai Atmaram Sagoon, Bombay.
9. Bharati Akshar Amba P. Kulkarni Vineet Chaitanya, 1996, Challenges in developing word analyzers for Indian languages, Presented at Workshop on Morphology, CIEFL, Hyderabad.
10. Bharati, , A., Sangal R., 1990, "A karaka based approach to parsing of Indian languages", proc of the 13<sup>th</sup> COLING vol 3, pp 30-35, Finland.
11. Bharati, Akshar and Rajeev Sangal, Parsing free wordorder languages using the Paninian framework, In ACL93: Proc.of Annual Meeting of Association for Computational Linguistics, Association for Computational Linguistics, New York, 1993a.
12. Bharati, Akshar, Dipti M Sharma, Vineet Chaitanya, Amba P Kulkarni, Rajeev Sangal, Durgesh D Rao, LERIL : Collaborative Effort for Creating Lexical Resources, In Proc. of

Workshop on Language Resources in Asian Languages, together with 6th NLP Pacific Rim Symposium, Tokyo

13. Bharati, Akshar, Rajeev Sangal and Vineet Chaitanya, Natural language processing, complexity theory and logic, In K.V.Nori and C.E. Veni Madhavan (eds.), Foundations of Software Technology and Theoretical Computer Science, Lecture Notes in Computer Science 472, Springer-Verlag, Berlin, 1991a, pp.410-420.
14. Bharati, Akshar, Vineet Chaitanya and Rajeev Sangal, 1995, Natural Language Processing: A Paninian Perspective, Prentice-Hall of India, New Delhi.
15. Bharati, Akshar, Vineet Chaitanya and Rajeev Sangal, A computational framework for Indian languages, Technical Report TRCS-90-100, Dept. of CSE, IIT Kanpur, July 1990b. (Course Notes for Intensive Course on NLP for Linguists, Vol.1)
16. Bharati, Akshar, Vineet Chaitanya and Rajeev Sangal, An appropriate strategy for machine translation in India. In S.S.Agarwal and Subhash Pani (eds.), Information Technology Applications in Language, Script and Speech, BPB Publications, New Delhi, 1994a.
17. Bharati, Akshar, Vineet Chaitanya and Rajeev Sangal, Course Material on A Computational Grammar based on Paninian Framework, Indian Society for Technical Education, Ministry of Human Resource Development, New Delhi, October 1993b.
18. Bharati, Akshar, Vineet Chaitanya and Rajeev Sangal, LFG, GB and Paninian Frameworks: An NLP viewpoint, In Tutorial on NLP at CPAL-2: UNESCO Second Regional Workshop on Computer Processing of Asian Languages, Dept. of CSE, IIT Kanpur, 12-16, March 1992b, pp. 1-42. (Also available as TRCS-92-140, Dept. of CSE, IIT Kanpur.)

19. Bhatta, V. P., 1991, 'Epistemology Logic and Grammar : Analysis of Sentence - Meaning', Eastern Book Linkers, Delhi
20. Bhattacharya, Ram Shankar, 1953, 'Kinds of agent as depicted by Pāṇini', Vak, Poona, 3: 129-33.
21. Blake, Barry J., 1994, 'Case', Cambridge Textbooks in Linguistics.: Cambridge University Press, Cambridge
22. Bobaljik, Jonathan and Collin Phillips, eds. 1993. Papers on Case and Agreement, Vol. 1. MIT Working Papers in Linguistics 19, Department of Linguistics, MIT.
23. Briggs, Rick, 1985, Knowledge representation in Sanskrit, *AI magazine*. Dr. Shivamurthy Swami, <http://www.taralabalu.org/panini/> (accessed: 30 April 2005).
24. Buhler, Georg, 1985, The Roots of the Dhatupatha not found in Literature, A Reader on the Sanskrit grammarians (J.F.Stall ed.), Motilal Banarasidass, Delhi.
25. Byrd, Roy, J., Calzolari, Nicoletta, Chodorow, Martin, S., Klavans, Judith, L., Neff, Mary, S., and Rizk, Omneya, A., 1987. Tools and Methods for Computational Linguistics (Computational Linguistics, 13:3-4, 219-240).
26. Cardona, George, 1964, 'Pāṇini's kāraṇa : agency, animation and identity', Journal of Indian Philosophy, Dordrecht. 2 : 231-306.
27. Cardona, George, 1967, 'Pāṇini's syntactic categories', Journal of the Oriental Institute, Baroda (JOIB) 16: 201-15.
28. Cardona, George, 1999, Recent Research in Pāṇinīan Studies, Motilal Banarasidass, New Delhi
29. Cardona, George, 2004, 'Some Questions on Pāṇini's Derivational system' In SPLASH proc. of iSTRANS, pp. 3.

30. Cardona, George, Pāṇini: his work and its traditions, Motilal banarasidass, New Delhi
31. Chomsky, Noam A., 1986a. Knowledge of Language: Its Nature, Origin and Use (MIT press, Cambridge, MA).
32. Date, C.J., Introduction to Database Systems, Addison-Wesley, Reading, MA, 1987.
33. Deshpande, Madhav M., 2002, Kāraṅkas: Direct and Indirect relationships, Indian Linguistic Studies estschrift in honor of George Cardona (Madhav M. Deshpande and Peter E. Hook ed.), Motilal Banarasidass, Delhi.
34. Desika, <http://tdil.mit.gov.in/download/Desika.htm> (accessed: 5 May 2005).
35. Divyadrusti, RCILTS, Utkal University, <http://www.its-utkal.org/ocrpage.htm> (accessed: 26 October 2005).
36. Dvivedi, H.P., 1978, 'Studies in Pāṇini' Inter-India Publications, Delhi.
37. Edgren, A. H., 1885, 'On the verbal roots of the Sanskrit language and of the Sanskrit grammarians' Journal of the Americal oriental Society 11: 1-55.
38. Fillmore, Charles J., The case for case, In E. Bach and R.T. Harms (eds.), Universals of Linguistic Theory, Holt Rinehart and Winston, New York, 1968, pp. 1-88.
39. G.V. Singh, Girish Nath Jha Indian theory of knowledge: an AI perspective proc. of seminar, ASR, Melcote, Mysore, 1994
40. Gillon, Brendan S., 2002, Bhartṛhari's Rule for Unexpressed Kāraṅkas, Indian Linguistic Studies festschrift in honor of George Cardona (Madhav M. Deshpande and Peter E. Hook ed.), Motilal Banarasidass, Delhi.
41. Giridharsharma Chaturvedi (ed.), 2004, Vaiyakaraṇasiddhāntakaumudī with Bālaṃanorama and Tattvabodhini ṭikā, Motilal Banarasidass, Delhi.

42. Grosz Barbara J., Jones Karen Sparck, Webber Bonnie Lynn. 1986: Readings in Natural Language Processing (Morgan Kaufmann Publishers, Inc, California)
43. Guru Prasad Shastri (ed). 1999, 'Vyakaraṇa-Mahābhāṣya with Pradipoddhat ṭikā' Pratibha Prakashan, Delhi.
44. Hart, George L., 1984, 'A Rapid Sanskrit method' Motilal Banarasidass, Delhi.
45. <http://www.cse.iitk.ac.in/users/langtech/anubharti.htm> (accessed: 30 November 2005).
46. <http://www-asia.human.is.tohoku.ac.jp/demo/vasia/html/> (accessed : 17 April 2005).
47. Huet, Gerard, <http://sanskrit.inria.fr/> (accessed : 30 January 2006).
48. Iyengar, D. Krishna, 1968, 'New Model Sanskrit grammar' The Samskrit Education Society, Madras.
49. Iyer, K.A.Subramania, 1969, 'Bhartṛhari : A study of the vākyapadīya in the light of ancient commentaries' Deccan College, Poona.
50. Jamie Joworski, Java 2 Platform unleashed, BPB Publication, Techmedia, Ansari Road, Dariya Ganj, New Delhi – 2.
51. Java Server Pages, <http://java.sun.com/products/jsp/> (accessed: 10 July 2006).
52. JAVA, Servlet, <http://java.sun.com/products/servlet/> (accessed: 10 July 2006).
53. Jha Girish N, 1993, Morphology of Sanskrit Case Affixes A Computational analysis Dissertation of M.Phil submitted to Jawaharlal Nehru University, New Delhi-110067.
54. Jha Girish N, 1994, Indian theory of knowledge: an AI perspective (proc. of national seminar on "Interface Mechanisms in Shastras and Computer Science", Academy of Sanskrit Research, Malcote, Mysore, April, 1994)

55. Jha Girish N, 1995, Proposing a computational system for Nominal Inflectional Morphology in Sanskrit (Proc. of national seminar on “Reorganization of Sanskrit Shastras with a view to prepare their computational database”, January, 1995)
56. Jha Girish Nath, 1996, Lexical conceptual structure and domain based machine translation system(jointly with Prof. Kapil Kapoor, Prof. G.V. Singh) presented at a symposium on Machine Aids for translation & communication, JNU, New Delhi-110067
57. Jha Girish Nath, 1999, English-SQL interface for databases (jointly with Prof. Jerry Morgan, head of the dept of linguistics at U of I, Urbana-Champaign) presented at an international conference on SALA held at University of Illinois
58. Jha Girish Nath, Mishra Sudhir K, Chandrashekar R, 2005, Information technology applications for Sanskrit lexicography: case of Amarakosha procs of the AsiaLex conference at National University of Singapore, Singapore
59. Jha Girish Nath, Mishra, S K, Chandrashekar R, Subash, August, 2005, developing a Sanskrit Analysis System for Machine Translation presented, at the National Seminar on Translation Today: state and issues, Deptt. of Linguistics, University of Kerala, Trivandrum.
60. Jha Girish Nath, November, 2003 A Prolog Analyzer/Generator for Sanskrit Subanta Padas Language in India
61. Jha Girish Nath, November, 2005 Language Technology in India: A survey Issue of C.S.I. magazine
62. Jha Girish Nath, February 2004, The System of Panini Language in India, volume 4:2

63. Jha Girish Nath, March 2004, Generating nominal inflectional morphology in Sanskrit  
SIMPLE 04, IIT-Kharagpur Lecture Compendium, Shyama Printing Works, Kharagpur,  
WB,
64. Jha Girish Nath,, Singh G.V., 2004, Astadhyayi Online Tata McGraw Hill, Proc of  
International Conference on Speech and Language Technology, New Delhi
65. Jha, Girish Nath, December, 2003, Current trends in Indian languages  
technology, Language In India, volume December.
66. Jijnasu, Pt. Brahmaddatt, 2003, 'Aṣṭādhyāyī Bhāṣya Vṛtti' Ramlal Kapoor Trust, Hariyana.
67. Joshi, Dayashankar Madhusudan, 1971, 'On expressing kāraka, a propos of Pāṇini 2.3.1'  
Indian Linguistics, Poona, 32: 107-12.
68. Joshi, S. D., 1962, 'Verbs and nouns in Sanskrit' Indian linguistics 32: 60-63.
69. Joshi, Shivaram Dattatray, 1974, 'Pāṇini's treatment of kāraka-relations', Festschrift  
Charudeva Shastri, pp. 258-70.
70. Joshi, Sivaram Dattatray & Roodbergen J. A. F., 1975, 'Patañjali's Vyākaraṇa  
Mahābhāṣya, Kāraḥnikā (P. 1.4.23-1.4.55)' CASS, University of Poona, Poona.
71. Joshi, Sivaram Dattatray & Roodbergen J. A. F., 1998, The Aṣṭādhyāyī of Pāṇini',  
Sahitya Akademi, New Delhi.
72. Jurafsky, Daniel and Martin H. James, Speech and languages processing, Pentice Hall,  
India
73. Jurafsky, Daniel, Speech and languages processing, Pearson Education Pvt.Ltd.,  
Singapore, 2005
74. Kale, M.R., 1972, 'A Higher Sanskrit grammar' Motilal Banarasidass, Delhi.

75. Kapoor, Kapil, 1985. Semantic Structures and the Verb : A propositional analysis (Intellectual Publications, New Delhi, 1985).
76. Kapoor, Kapil, 1996. Panini's derivation system as a processing model (to appear in the proc. of "A Symposium on Machine Aids for Translation and Communication, 11-12 April, School of Computer & Systems Sciences, J.N.U. New Delhi, 1996)
77. Katre, S.M., 1968, 'Dictionary of Pāṇini' Deccan College, Poona.
78. Katz, J. J., 1972. Semantic Theory (Harper and Row, New York).
79. Kielhorn, F., 1970, 'Grammar for Sanskrit Language' Chowkhamba Sanskrit Series office, Varanasi.
80. Kiparsky, P. and Stall, J. F., 1969. Syntactic and Semantic Relation in Panini (Foundations of Language, Vol.5, 83-117).
81. Kumar Sachin and Jha Girish Nath, December 2005 A Paninian Sandhi analyzer for Sanskrit in the proc. at Platinum Jubilee International Conference, L.S.I. at Hyderabad University, Hyderabad page-35
82. Macdonell, A., 1979, 'Sanskrit grammar for students' Motilal Banarasidass, Delhi.
83. Mahavir, 1984, 'Samartha theory of Pāṇini and sentence derivation' Munshiram Manoharlal Publishers Pvt. Ltd., Delhi.
84. Mishra Sudhir K, Jha Girish Nath, 2004, A karaka analyzer for Sanskrit Tata McGraw Hill, 2004, Proc of International Conference on Speech and Language Technology, New Delhi
85. Mishra Sudhir K, Jha Girish Nath, 2005, Identifying verb inflections in Sanskrit morphology Proc. of SIMPLE05, IIT Kharagpur



86. Mishra Sudhir K. Jha Girish Nath, December, 2005, Karaka analysis of complicated Sanskrit sentences in the proc. at Platinum Jubilee International Conference, L.S.I. at Hyderabad University, Hyderabad page-34
87. Mishra, Shrinarayana (ed.), 1996, 'Kāśikā with prakāś commentary', Chawkhambha Sanskrit Sansthan, Varanasi.
88. Mukharjee, Bendre and Sangal, Natural languages processing, Proceeding, ICON, 2004
89. Mukharjee, Bendre and Sangal, Natural languages processing, Proceeding, ICON, 2005
90. Muller, F. Max, 1983, 'A Sanskrit grammar' Asian Educational Services, Delhi.
91. Murti, M. Srimannarayana, 1997, 'Bhartrhari the Grammarian' Sahitya Akademi, Delhi.
92. Narayan Mishra(ed). Kashika of Pt.Vamana and Jayaditya ,Chaukhamba Sanskrit sansthan, Varanasi,1996.
93. Narayana, V. N., 1994.Anusaraka: a device to overcome the language barrier (Ph.D. thesis, Deptt of CSE, IIT, Kanpur, India)
94. Nautiyal, Chakradhar Hans, 1995, 'Bṛhada-anuvād-candrikā' Motilal Banarasidass, Delhi.
95. Nomiya, H., 1992.Machine Translation by case generation (proc. of IV International Conference on Computational Linguistics, Nantes, France, 714-720)
96. Norvig, Russell, Artificial Intelligence: A modern approach, Pentic Hall, India
97. Ostler, N. D. M. (1979). Case-linking: a theory of case and verb diathesis applied to classical Sanskrit: 434 leaves.
101. Paranjpe, Vinayak W.,1957, ' Analysis of case suffixes with special reference to Pāṇini's grammar', Akten des... international Orientalisten-Kongresses(AIOK) 24:574-77.

102. Pen to Paper, The House Magazine of C-DAC, <http://www.cdacindia.com/html/adp/mactrans.asp> (accessed: 10 February 2006).
103. Pen to Paper, The House Magazine of C-DAC, <http://www.cdacindia.com/html/connect/3q2000/art10a.htm#> (accessed: 10 March 2006).
104. P. Navrat and H Ueno (eds), Knowledge Based Software Engineering IOS Press, Amsterdam, Netherlands, 1998.
105. Peter M. Scharf and Malcolm D. Hyman, <http://sanskritlibrary.org/morph/> (accessed: 12 March 2006).
106. Peter M. Scharf and Malcolm D. Hyman, <http://sanskritlibrary.org/> (accessed: 17 March 2006).
107. Peter Naur, Brian Randell, J N Buxton (eds.), Software engineering : concepts and techniques, Mason/Charter Publisher Inc, New York, 1976
108. Pt.Brahmadatta jigyasu (ed.), 1998, 'Panini-Astadhyayi', Ramlal kapoor trust, Sonipat.
109. R.M.K. Sinha, 1989, "A Sanskrit based Word-expert model for machine translation among Indian languages", Proc. of workshop on Computer Processing of Asian Languages, Asian Institute of Technology, Bangkok, Thailand, Sept.26-28, 1989, pp. 82-91.12.
110. Rajeev Sangal, <http://www.elitexindia.com/paper2004/rajeevsangal.pdf> (accessed : 7 April 2006).
111. Ramakrishnamacharyulu, K.V., Paninian Linguistics and Computational Linguistics, Samvit, Series no. 27. Pp. 52-62, Academy of Sanskrit Research, Melkote, Karnataka (India), 1993.

112. RCILTS, School of Computer & System Science, <http://rcilts.jnu.ac.in> JNU, New Delhi. (Accessed: 17 April 2006).
113. RCILTS, Utkal University, <http://www.its-utkal.org/nlppage.htm> (accessed : 20 April 2006).
114. Reyle, U. and C. Rohrer (eds.), Natural Language Parsing and Linguistic Theories, D. Reidel, Dordrecht, 1988.
115. Rocher, R., 1964, ‘‘Agent’ et ‘object’ chez Pāṇini’ JAOS 84, pp 44-54
98. Sanskrit academy <http://www.sanskritacademy.org/Achievements.htm> (accessed: 7 April 2005).
116. Sanskrit, compound, <http://www.sanskrit-sanscrito.com.ar/english/sanskrit/sanskrit8intro.html#Copulativecompounds> (accessed: 30 May 2006).
117. Scharf Peter M., 2002, Pāṇini, Vivakṣā and Kāraka-rule-ordering, Indian Linguistic Studies festschrift in honor of George Cardona (Madhav M. Deshpande and Peter E. Hook ed.), Motilal Banarasidass, Delhi.
118. Shabdabodha, TDIL, Gov. of India, <http://tdil.mit.gov.in/download/Shabdabodha.html> (accessed: 22 April 2006).
119. Sharma, K. Madhava Krishna, 1968, ‘Pāṇini Patañjali and Kātyāyana’ Shri Lal Bahadur Shastri Rashtriya Sanskrit Vidyapeeth, Delhi.
120. Sharma, Rama Nath, 2003, ‘the Aṣṭādhyāyī of Pāṇini’, Munshiram Manoharlal Publishers Pvt. Ltd., Delhi.
121. Shastri Charu Deva, 1991, ‘Pāṇini : Re-interpreted’ Motilal Banarasidass, Delhi.

122. Singh, Jag Dev and K. Doraswamy, 1972, 'The case: Tolkā ppiyam and Pāṇini, a comparative study', Kurukshetra University Research Journal 4: 119-29.
123. Singh, Jag Dev, 1974, 'Pāṇini's theory of kārakas', International Journal of Dravidian linguistics, Trivandrum, 3:287-320.
124. Sinha, Anil C., 1973, 'Generative semantics and Pāṇini's kāraka', Journal of the Oriental Institute, Baroda (JOIB) 23: 27-39.
125. Subash and Jha Girish Nath, December 2005, Morphological analysis of nominal inflections in Sanskrit in the proc. at Platinum Jubilee International Conference, L.S.I. at Hyderabad University, Hyderabad page 34
126. TDIL, <http://tdil.mit.gov.in/mat/ach-mat.htm> (accessed: 10, November 2005).
127. TDIL, <http://tdil.mit.gov.in/TDIL-OCT-2003/machine%20translation%20system%20.pdf> (accessed: 20 November 2005).
128. The Web server, Apache Tomcat, <http://www.apache.org/> (accessed: 12 July 2006).
129. Vivek Quarterly journal on Artificial Intelligence, NCST, Mumbai, <http://www.ncst.ernet.in/kbcs/vivek/issues/13.2/sumam/sumam.html> (accessed: 27 April 2006).
130. Whitney, W.D., 2002, 'History of Sanskrit grammar' Sanjay prakashan, Delhi.
131. Whitney, William Dwight, 1983, 'Sanskrit Grammar' Motilal Banarasidass, Delhi.
132. Whitney, William Dwight, 1983, 'The Root verb-forms and Primary derivatives of the Sanskrit language' Motilal Banarasidass, Delhi.
133. Yogi, Satyabhushana & Shashikumar, 1985, 'Nighaṇṭu tathā nirukta', Motilal Banarasidass, Delhi.



