Computational Methods for Anaphora and Cataphora Resolution in the Sanskrit Text *Panchatantra*

Dissertation submitted to
Jawaharlal Nehru University
in partial fulfilment of the requirements
for the award of the Degree of
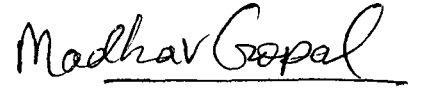
MASTER OF PHILOSOPHY

MADHAV GOPAL

Centre for Linguistics
School of language, Literature & Culture Studies
Jawaharlal Nehru University
New Delhi-110067
INDIA
2011

Date 21st July, 2011

<u>DECLARATION BY THE CANDIDATE</u>

This dissertation titled **"Computational Methods for Anaphora and Cataphora Resolution in the Sanskrit Text *Panchatantra*"** submitted by me for the award of the degree of Master of Philosophy, is an original work and has not been submitted so far in part or in full, for any other degree or diploma of any University or Institution.

(MADHAV GOPAL)
M. Phil. Student
CL/SLL&CS
JNU

i

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

Dated: 21st July 2011

CERTIFICATE

This dissertation titled **"Computational Methods for Anaphora and Cataphora Resolution in the Sanskrit Text *Panchatantra"*** submitted by **Madhav Gopal** to the Centre for Linguistics, School of Language, Literature and Culture Studies, Jawaharlal Nehru University, New Delhi, for the award of the degree of **Master of Philosophy**, is an original work and has not been submitted so far in part or in full, for any other degree or diploma of any University or Institution.

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

(PROF. ANVITA ABBI)   (DR. GIRISH NATH JHA)   (PROF. FRANSON D. MANJALI)

Dr. GIRISH NATH JHA

SUPERVISOR   CO-SUPERVISOR   CHAIRPERSON

Professor Anvita Abbi
Centre for Linguistics
School of Language, Literature & Culture Studies
Jawaharlal Nehru University
New Delhi - 110 067

Special Centre for Sanskrit Studies

To My Teachers and Parents

*ātmā vā are draṣṭavyaḥ śrotavyo mantavyo nidid*h*yāsitavyo maitreyyātmano vā are darśanena śravaṇena matyā vijñānenedaṃ sarvaṃ viditam.*

"Verily, it is one's self (*ātman*), O Maitreyi, which one should see, hear, on which one should reflect and ponder. For by seeing and hearing one's self, by reflecting and pondering on one's self, one gains the knowledge of the whole world."

(Bṛhadāraṇyaka Upaniṣad 2.4.5)

Acknowledgement

I am happy to acknowledge the generous help I have received in working on this dissertation. First of all, I would like to express my gratitude to my mentor and supervisor Prof. Anvita Abbi for her continuous help, encouragement and perceptive comments on the material in the earlier drafts of this dissertation which greatly helped me to improve upon. Despite her busy schedule she always gave me her wise suggestions whenever I needed it. Without her all kinds of support and motivation, it would have been extremely difficult for the present work to come in this form.

I am thankful to my co-supervisor Dr. Girish Nath Jha, who is credited to motivating me to work on computational linguistics along with Sanskrit. He initiated me in this field since my MA days at JNU and trained many of us in Java programming and developing database, creating text files and other linguistic resources. Without him, this undertaking could never have completed. It was he who showed and provided most of the programming part of this work, I being the person who presented the algorithm and the structure of the language.

I am grateful to all my teachers at the Centre for Linguistics, School of Language, Literature and Culture Studies at JNU from whom I learnt the basics of linguistics. I express my deepest thanks and gratitude to Prof. Franson D. Manjali, Chairperson of the Centre for Linguistics, from whom I learnt a bit of philosophy of language, Prof. P.K.S. Pandey who taught me phonetics and phonology, Dr. Ayesha Kidwai from whom I could learn some generative syntax, Prof. Vaishna Narang for initiating me in the area of applied linguistics, Asst. Prof. Hari Madhab Ray who taught me generative morphology and helped a lot when I was beginning linguistics and Ms. Yankee Modi for providing me some understanding of sociolinguistics and field linguistics. I also acknowledge the office staff of the centre especially Rawat ji and Navin ji for their cooperation on various occasions. I am thankful to Prof. K.V. Subbarao (DU) for giving me his insightful suggestions and for various helps. I gratefully extend my humble thanks to my teachers at University of Allahabad Prof. L.R. Sharma, Prof. Ram Kishor Shastri, Dr. Uma Kant

v

Contents

x

Abbreviations Used

1	First Person
2	Second Person
3	Third Person
ABL	Ablative
ACC	Accusative
AUX	Auxiliary
CAUS	Causative
COMP	Complementizer
COND	Conditional
CONJ	Conjunction
CORR	Correlative
DAT	Dative
DECL	Declarative
DEF	Definite marker
DEM	Demonstrative
DET	Determiner
DIS	Disjunction
DM	deictic marker
DST	Distal
DU	Dual
EMPH	Emphatic marker
EXPL	Expletive
F	Feminine
FUT	Future
GEN	Genitive
GRN	Gerund
HON	Honorific

IMP	Imperative
IMPF	Imperfect
INDEF	Indefinite marker
INF	Infinitive
INS	Instrumental
JSP	Java Server Pages
LOC	Locative
M	Masculine
NEG	Negative
N	Neuter
NOM	Nominative
PST	Past
PERF	Perfect
PCPL	Participle
PSPL	Past participle
PRPL	Present participle
PL	Plural
POT	Potential (*vidhiliṅga*)
POS	Part Of Speech
POSS	Possessive
PRX	Proximate/Proximal
PASS	Passive
POSS	Possessive
PRS	Present
PT	*Pañcatantra*
QUOT	Quotative
REL	Relative
REC	Reciprocal
REFL	Reflexive
SARS	Sanskrit Anaphora Resolution System

SG	Singular
SUBJ	Subject
VOC	Vocative
VR	Verb Root
mtbd	mitrabhedam
mtsp	mitrasamprāptikam
kkly	kākolūkīyam
ldpr	labdhapraṇāśam
apkr	aparīkṣitakārakam

Transcription Key Used in the Dissertation

अ = a

आ = ā

इ = i

ई = ī

उ = u

ऊ = ū

ऋ = ṛ

ए = e

ऐ = ai

ओ = o

औ = au

ं (anusvāra) = ṃ

ः (visarga) = ḥ

क् = k

ख् = kʰ

ग् = g

घ् = gʰ

ङ् = ṅ

च् = c

छ् = cʰ

ज् = j

झ् = jʰ

ञ् = ɲ

ट् = ṭ

ठ् = ṭʰ

ड् = ḍ

ढ् = ḍʰ

ण् = ṇ

त् = t

थ् = tʰ

द् = d

ध् = dʰ

न् = n

प् = p

फ् = pʰ

ब् = b

भ् = bʰ

म् = m

य् = y

र् = r

ल् = l

व् = v

श् = ś

ष् = ṣ

स् = s

ह् = h

क्ष् = kṣ

त्र् = tr

ज्ञ् = jɲ

xiv

List of Tables and Figures

Tables

Figures

Chapter 1
Introduction

This chapter discusses the issues involved in the anaphora resolution of natural languages like Sanskrit. The chapter also introduces fundamental ideas on Sanskrit, *Pañcatantra* (PT), anaphors and anaphora resolution theories etc. At the end of the chapter a summary of the remaining chapters has been given.

1.1. Sanskrit and *Pañcatantra*

1.1.1. Sanskrit

Sanskrit has a position in India and Southeast Asia similar to (or rather more than) that of Latin and Greek in Europe, and is a central part of Hindu tradition. It is one of the 22 official languages of India and a state language of Uttarakhand. Sanskrit is one of the well-studied languages of the world, having a sophisticated vocabulary, morphology, literature, research, scholarship and most importantly a rich grammatical tradition. The ancient knowledge of Indian subcontinent is stored in the Sanskrit texts and it needs to be explored today to benefit the humanity. Plenty of literature is available on the highly philosophical and ethical subjects, various sciences, linguistic investigations and other academic disciplines. The linguistic investigation of this language was focused on the morphology and phonology, describing it variously. However, its syntax has received least attention by linguists. Mainly, Speijer (1886), Delbruck (1888), and Hans H. Hock (1991) could be counted among those who have been interested in syntactic structure of this language.

1.1.2. *Pañcatantra*

Pañcatantra is an old text of ethical stories believed to be written by Vishnu Sharma. The 72 stories in five (*pañca*) sections (*tantras*) are replete with some moral lesson, not only for children but also for the humans in general. This is a *nīti grantha* (ethical text) and it is, to a great extent, relevant today as well. It aims to make us learn how we can use the

1

knowledge which we have already acquired and how we should behave in the society and how we should understand the society. The five *tantras* are *mitrabhedam* (the separation of friends), *mitrasamprāptikam* (the acquisition of friends), *kākolūkīyam* (about the crows and the owls), *labdhapraṇāśam* (the loss of what was gained), and *aparīkṣitakārakam* (the rash doer). In *mitrabheda* it is described how differences and enmity can be created between two closed friends. The lion Pingalaka and the bull Sanjeevaka were closed friends. The two jackals Karataka and Damanaka got them separated and Sanjeevaka was killed by Pingalaka. In *mitrasamprāptikam* the teaching is that one should have many useful friends. The crows, tortoises, deer, and rats lived happily due to their closed friendship despite the scarcity of various resources. The *kākolūkīyam* is about deceitful politics in which teaching is that for one's gain one should befriend their enemy and then deceivingly destroy them. In the story the crow befriends the owl and then set owl's fort on fire. In *labdʰapraṇāśam* the teaching is that a wise person wins the battle and a fool looses even things on his own hands. A monkey and a crocodile enter in a friendship. The wife of the crocodile wants the tasty heart of the monkey. The monkey saves himself by saying that he has left his heart in the tree, so he be taken to the bank of the river. The monkey runs away and the crocodile is embarrassed. The teaching of the *aparīkṣitakārakam* is that one should do things after a great consideration. A Brāhmaṇi kills her mongoose that saved her child from the snake, on the suspicion that the mongoose has killed the child. Later she repents for her misdeed.

The central theme of the PT is the harmonious and integrated development of a man, a life in which security, prosperity, friendship and learning are combined as to produce a lasting joy. It exemplifies and upholds ethical values, social order, and customary law and yet is an important medium through which protest, dissent and reform are articulated. Vishnu Sharma, a great pundit of *loka* (world) and *śāstra* (academic knowledge), tries his best to convey his teachings to his pupils through the animal characters. He successfully made his pupils practical, good-behaved, and diplomatic in a short period of time. This work was so popular that even in ancient times the translations of this text were available in foreign languages in foreign lands. The purpose behind writing the PT was to implant

2

moral values and governing skills in the young sons of the ancient kings. The ancient Sanskrit text boasts of various animal stories in verse and prose.

The style of PT is simple and proverbial. The description of the theme is written in prose while the aphorisms are composed in verses. According to Baladeva Upadhyaya (2001) these verses are taken from the Ramayana, Mahabharata, and other ancient ethical texts. The production of PT certifies it to be the text of politics and worldly behaviour.

According to Baladev Upadhyay (2001), there are four recensions of PT available now.The first one is the Pahlavi translation, though it is not available but the information about its stories is received from Syraic and Arabic translations. The second recension is gone within the *Bṛhatkathā* of Guṇāḍhya. This *Bṛhatkathā* was in Paiśāci language which was originally lost but its translations in the form of the *Bṛhatkatʰāmaṇjari* of Kṣemendra and the *Kathāsaritsāgara* of Somadeva are available now. The third recension is called *tantrākhyāyikā* with which *Jainkathāsaṃgraha* is also related. The *tantrākhyāyikā* is the oldest recension and its original place is Kashmir. Hertel believes that *tantrākhyāyikā* is the original form of the PT available today. The fourth one is southern PT and the Nepali PT and *Hitopadeśa* are its representatives. Thus the PT is not a common text but a representative of a plenty of literature (Upadhyay, 2001).

1.1.2.1. *The Timing of the Pañcatantra*

There is a lack of consensus among scholars regarding the time of the PT. The German scholar Hertel believes its creation after 200 BCE, whereas Keith believes it to be around 200 AD or afterwards. The assumption of Keith is based on the use of the word *Dinar* in the text. Dinar was a Roman coin that was brought India around 100 AD. Baladev Upadhyay (2001) says that it would not be improper to assume the timing of PT between 300 AD and 400 AD.

1.1.2.2. *The POS Tagging of the Pañcatantra*

This text has been POS tagged with Indic Language POS Tagset (IL-POST) developed by Microsoft Research India (MSRI) by the author of this dissertation and the data have been released by Linguistic Data Consortium (LDC) (It can be reached at

3

http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2011T04). The title of this corpus is *Indian Language Part-of-Speech Tagset: Sanskrit*. Its Linguistic Data Consortium (LDC) catalog number is LDC2011T04 and ISBN is 1-58563-575-8. This is a corpus developed by MSRI India to support the task of Part-of-Speech Tagging (POS) and other data-driven linguistic research on Indian Languages in general. It is created as a part of the Indian Language Part-of-Speech Tagset (IL-POST) project, a collaborative effort among linguists and computer scientists from MSR India, AU-KBC (Anna University, Chennai), Delhi University, IIT Bombay, Jawaharlal Nehru University (Delhi) and Tamil University (Tamil Nadu).

The Linguistic Data Consortium is an open consortium of universities, companies and government research laboratories. It creates, collects and distributes speech and text databases, lexicons, and other resources for research and development purposes. The University of Pennsylvania is the LDC's host institution. The LDC was founded in 1992 with a grant from the Advanced Research Projects Agency (ARPA), and is partly supported by grant IRI-9528587 from the Information and Intelligent Systems division of the National Science Foundation.

1.1.2.3. *The POS Tagging Scheme of the IL-POST*

The tagset used for tagging PT, was derived from IL-POST, a standard framework for Indian languages developed by MSRI. This is a hierarchical tagset which is based on guidelines similar to EAGLES. This tagset encodes information at three levels- Categories, Types, and Attributes. Till date (as far as we know) four language specific tagsets have been derived from this framework; Hindi, Bangla, Tamil and Sanskrit.

The structure of the tagset is made up of 9 major categories: noun, verb, pronoun, nominal modifier, demonstrative, *kridanta*, particle, punctuation, and residual. These categories have their respective types and then attributes. Noun is given two types-common and proper and their attributes are gender, number, case and nominal declension. Verb has no types but has pada, number, person, tense/mood and honorificity as attributes. Pronouns are divided in 5 sub categories and have gender, number, person,

4

case and nominal declension. Nominal Modifier has two types- adjective and quantifier. Adjectives are given gender, number, case, nominal declension, emphatic, negative and honorificity attributes. Quantifiers' attributes are same as adjectives except honorificity (don't have) and numeral (have). Demonstrative Category has three types- absolute, relative, and wh. Their attributes are gender, number, person, case, nominal declension, distance and honorificity. The category of *kridanta* has participle and gerundive as types and gender, number, case, nominal declension as attributes. Particles are divided in 13 types. The final two categories Punctuation and Residual (foreign word, symbol, others) don't have any attribute. The Appendix 2 contains the tagset.

The standard which has been followed in this tagset takes care of the linguistic richness of Indian languages. It allows the "selective inclusion and removal of features for a specific language/project, thereby keeping the framework a common standard across languages/projects". This tagset is supposed to provide cross-linguistic compatibility, reusability, and interchangeability. Keeping in mind its higher accuracy results, it could prove a boon to Indian NLP which still needs to achieve its good position in the world as Indian languages are lacking adequate resources in terms of data and tools (Baskaran et al. (2008) and Girish et al. (2009)).

This framework is very sophisticated in covering the linguistic features of Indian languages. The tags used are extremely fine-grained, and incorporate a great deal of information about case, gender, number and so on. The IL-POST is able to accommodate all desired linguistic features of Sanskrit and the tagged corpus remains compatible with other languages tagged by a brethren tagset. Apart from this as this tagset follows the standard of European languages it could enable us to access the European language data for Indian NLP and vice versa.

MSRI has developed an annotation tool called *MSRI Part-of-Speech Annotation Interface* which is a GUI for assigning the appropriate POS label for each word in a sentence. This is designed for annotating within the framework of a hierarchical tagset designed at

5

MSRI. The tool also provides the facility to provide the morphological attributes with their values. The interface supports various operations for viewing and editing the POS labels and morphological information associated with the word. This tool reduces the cognitive load of the human annotator and enables one to speed up the tagging task.

1.2. Preliminaries on Anaphora Resolution

Anaphora is a Greek word and etymologically it means 'the act of carrying back upstream'. In linguistics "anaphora is cohesion (presupposition) which points back to some previous item" (Mitkov, 1996). The 'pointing back' (reference) is called an anaphor and the entity to which it refers is its antecedent. In (1) 'she' is an *anaphor* and 'Manisha' is its *antecedent*. 'She' cannot be interpreted without pointing back to 'Manisha'.

(1) Manisha is staying at Allahabad. She is a professor of Sanskrit.

There is also a slightly different device which languages use, sometimes, to create a sense of mystery or expectancy. This device is called cataphora. In (2) 'she' is pointing forward to 'Prof. Vibha' to derive its meaning. So, in this case 'she' is a *cataphor* and the process of referring forward is called *cataphora*.

(2) She is a person of international repute, widely known for her excellent research and scholarship. Prof. Vibha has been awarded the prestigious Ambedakar Professorship at the University of Allahabad.

The process of determining the antecedent of an anaphor is called anaphor resolution. The term 'anaphora resolution' is very common among computational linguists; however, Mitkov (1996) considers it less accurate than 'anaphor resolution'. Anaphora is a linguistic phenomenon: what is resolved is therefore not the phenomenon but the anaphor (reference) which is initially 'unknown' and whose antecedent must be tracked down.

Anaphora resolution is a challenging problem in Natural Language Processing (NLP) and plays a significant role in most of the NLP applications such as machine translation (MT), text summarization, question answering systems, information extractions, etc. In

6

Sanskrit, however, this has received very little computational research. Until now, only a few papers are available on this topic. Anaphora resolution is a very wide topic and generally, it includes lexical anaphors, third person pronouns, and other anaphoric items in NLP.

For humans, it is relatively easy to determine the antecedents of anaphors in the text but machines have immense problems in identifying them. The proposed study aims at studying Sanskrit pronouns, lexical anaphors (reflexives and reciprocals) and cataphors in the ethics text of PT and designing a computational tool called *Sanskrit Anaphora Resolution System* (SARS) to resolve the lexical anaphors and cataphors automatically.

1.3. Anaphora Resolution: Various Approaches

Anaphora resolution has been the key topic of NLP research for over a couple of decades. Plenty of literature is available discussing problems and solutions for pairing anaphors and antecedents. In the literature on anaphora resolution, mostly, third person pronouns, zero pronouns and lexical anaphors (reflexive and reciprocal pronouns) are dealt with. The anaphora resolution algorithms can be divided into knowledge rich and knowledge poor algorithms. The knowledge rich algorithms are the approaches which need a deeper level of analysis such as parsed sentences with grammatical roles specified. On the other hand, knowledge poor algorithms just need a shallow linguistic analysis. Following are some of the prominent anaphora resolution theories.

1.3.1. Lappin and Leass Approach

Lappin and Leass Algorithm (1994) or RAP (*Resolution of Anaphora Procedure*), aims at identifying both intra and intersentential antecedents of third person pronouns and lexical anaphors (reflexive and reciprocal), in English. This algorithm is based on salience heuristics. RAP employs a simple weighting scheme that integrates the effects of the recency and syntactically-based preferences; but no semantic preferences are employed beyond those forced by agreement. Following are the main components of the Lappin and Leass Algorithm:

7

- An intrasentential syntactic filter for ruling out anaphoric dependence of a pronoun on a noun phrase, based on syntactic grounds;

- A morphological filter for ruling out anaphoric dependence of a pronoun on a noun phrase due to person, number, or gender non-agreement;

- An anaphor binding algorithm for identifying the possible antecedent binder of a lexical anaphor (reciprocal or reflexive pronoun) within the same sentence;

- A procedure for assigning the suitable salience factors weights to a noun phrase, according to its grammatical role, such as syntactic parallelism, subject, etc.;

- A decision procedure for selecting the preferred element from a list of possible antecedent candidates.

The anaphor binding algorithm identifies the intrasentential candidates for reflexive or reciprocal pronouns; the syntactic filter rules out the intrasentential coreference candidates for third person pronouns that are unlikely to be the antecedent. For the remaining candidates, the value of the salience factors is calculated. The chosen referent will be the one with the highest salience factor. When there is more than one candidate with the same salience factor, the algorithm chooses the candidate which is closer to the pronoun. The syntactic filter and the anaphor binding algorithm analyze the pronoun's sentence syntactic structure to decide if co-reference is allowed. The algorithm uses the grammatical representation generated by Slot grammar parser developed by McCord (1990). The Slot grammar is a full parser and relies on salience measures derived from the syntactic structure and a simple dynamic model of attentional state to select the antecedent noun phrase of a pronoun from a list of competing candidates.

RAP uses a salience weighting system based on syntactic features. There are two types of operations performed by the algorithm: discourse model update and pronoun resolution. When a noun phrase, which introduces a new entity in the discourse, is found, a representation for that entity is created and its salience factor is calculated. The salience factor for a given entity is the total of all salience factors applied to that entity. The initial salience factors are presented in the table below.

8

Salience factors	Weights
Sentence recency	100.0
Subject emphasis	80.0
Existential emphasis	70.0
Accusative (direct object) emphasis	50.0
Indirect object and oblique complement emphasis	40.0
Non-adverbial emphasis	50.0
Head noun emphasis	80.0

Table 1: Salience factors with initial weights (Lappin and Leass, 1994)

The salience factor weights show the preference for choosing a certain antecedent based on its grammatical role, according to the following hierarchy (Jurafsky and Martin, 2000):

subject > existential predicate nominal > object > indirect object or oblique > demarcated adverbial PP

The algorithm was tested on computer manual texts. A blind test was also conducted on manual text containing 360 pronoun occurrences. According to the claim the algorithm successfully identified 86% of these pronouns.

The algorithm has the limitation that it did not have any real world knowledge and also it did not model intentional or global discourse structure.

1.3.2. Tree Search Algorithm

Hobbs (1978) describes an algorithm for pronoun resolution which takes syntactic representations of the sentences up to and including the current sentence as input and

9

performs a search for an antecedent noun phrase on these trees. There is no explicit representation of a discourse model or preferences as in the Lappin and Leass algorithm. However, certain of these preferences are approximated by the order in which the search on syntactic trees is performed.

An algorithm that searches parse trees must also specify a grammar, since the assumptions regarding the structure of syntactic trees will affect the results. A fragment for English that the algorithm uses is given in the following figure.

S → NP VP
NP → { (Det) Nominal ({PPor Rel})*}
Det → {determiner or NP's}
PP → preposition NP
Nominal → noun (PP)*
Rel → wh-word S
VP → verb NP (PP)*
Table 2: A grammar fragment for the Tree Search algorithm.

The steps of the algorithm are as follows:

1. Begin at the noun phrase (NP) node immediately dominating the pronoun.

2. Go up the tree to the first NP or sentence (S) node encountered. Call this node X, and the path used to reach it p.

3. Traverse all branches below node X to the left of path p in a left-to-right, breadth-first fashion. Propose as the antecedent any NP node that is encountered which has an NP or S node between it and X.

4. If node X is the highest S node in the sentence, traverse the surface parse trees of previous sentences in the text in order of recency, the most recent first; each tree is traversed in a left-to-right, breadth-first manner, and when an NP node is encountered, it is proposed as antecedent. If X is not the highest S node in the sentence, continue to step 5.

10

5. From node X, go up the tree to the first NP or S node encountered. Call this new node X, and call the path traversed to reach it p.

6. If X is an NP node and if the path p to X did not pass through the Nominal node that X immediately dominates, propose X as the antecedent.

7. Traverse all branches below node X to the left of path p in a left-to-right, breadth-first manner. Propose any NP node encountered as the antecedent.

8. If X is an S node, traverse all branches of node X to the right of path p in a left-to-right, breadth-first manner, but do not go below any NP or S node encountered. Propose any NP node encountered as the antecedent.

9. Go to Step 4.

To test the algorithm Hobbs took a text from an archaeology book, an Arthur Hailey's novel and a copy of Newsweek. From each of these as much contiguous text as was necessary to obtain one hundred occurrences of pronoun was taken. His algorithm worked successfully for 88% of the time, and 92% when augmented with simple selectional constraints. Clearly the algorithm by itself is inadequate. However, Hobbs suggests that it may still be useful, as it is computationally cheap compared to any semantic method of pronoun resolution.

1.3.3. The Centering Theory

The centering theory developed by Grosz, Joshi, Weinstein (1995) claims that certain entities mentioned in an utterance are more central than others and this property imposes constraints on speaker's use of different types of referring expression. This theory relates focus of attention, choice of referring expression and perceived coherence of utterance within a discourse segment.

The term 'center' is used for an utterance to refer to those entities serving to link that utterance to other utterances in the discourse segment that contains it. It is an utterance and not a sentence in isolation that has centers. The same sentence uttered in different

discourse segments may have different centres. Centres are thus discourse constructs. Furthermore, centres are semantic objects, not words phrases or syntactic forms.

Each utterance U in a discourse segment (DS) is assigned a 'forward looking centre' $C_f(U, DS)$. Each utterance other than the segment initial utterance is assigned a single 'backward looking centre' $C_b(U, DS)$. These centres are sometimes also denoted as $C_f(U)$ and $C_b(U)$.

The backward looking centre of utterance U_{n+1} connects with one of the forward looking centres of utterance U_n. In fact, $C_b(U_{n+1})$ is by definition the most highly ranked element of $C_f(U_n)$. Regarding the ordering of the elements in $C_f(U_n)$ the following hierarchy is used.

> subject > existential predicate nominal > object > indirect object or oblique > demarcated adverbial PP

In centering based algorithms there are no numerical weights attached to the entities of the list. They are simply ordered relative to each other. The highest ranked forward looking center is called Cp (preferred center).

There are various algorithms devised for anaphora resolution which are centering based. Brennan et al. (1987, henceforth BFP), Walker et al. (1994), Kameyama (1986) and Strube & Hahn (1996) are some of them. In BFP algorithm preferred referents of pronouns are computed from the relations that hold between the forward and backward looking centers in adjacent sentences. Four intersentential relationships between a pair of utterances U_n and U_{n+1} are defined which depend upon the relationship between $C_b(U_{n+1})$, $C_b(U_n)$, and $C_p(U_{n+1})$; these are shown in the following figure.

	$C_b(U_{n+1}) = C_b(U_n)$ or undefined $C_b(U_n)$	$C_b(U_{n+1}) \neq$ $C_b(U_n)$
$C_b(U_{n+1}) = C_p(U_{n+1})$	Continue	Smooth-Shift
$C_b(U_{n+1}) \neq C_p(U_{n+1})$	Retain	Rough-Shift

Table 3: Transitions in the BFP algorithm

The following rules are used by the algorithm:

- Rule 1: If any element of $C_f(U_n)$ is realized by a pronoun in utterance U_{n+1}, then $C_b(U_n)$ must be realized as a pronoun also.

- Rule 2: Transition states are ordered. Continue is preferred to Retain, Retain is preferred to Smooth-Shift, Smooth-Shift is preferred to Rough-Shift.

The algorithm has been defined as follows:

1. Generate possible C_b-C_f combinations for each possible set of reference assignments.

2. Filter by constraints, e.g., syntactic coreference constraints, selectional restrictions, centering rules and constraints.

3. Rank by transition orderings.

The pronominal referents that get assigned are those which yield the most preferred relation in rule 2, assuming that Rule 1 and other Coreference constraints (gender, number, syntactic, selectional restrictions) are not violated.

1.3.4. Earlier Works on Sanskrit Anaphora Resolution

The development and design of SARS have derived benefits from the existing research of anaphora resolution for Sanskrit. The algorithms proposed in earlier works have been studied before finalizing our algorithm. Girish Nath Jha et al. (2008 and 2009) have done a wider case study of anaphors in Sanskrit scanning the language from the epics to *Shivarajavijayam*. They have discussed about the concept of anaphora and anaphora

resolution techniques in Indian intellectual tradition including Vyākaraṇa, Nyāya, and Mīmāṃsā. The papers have focused on pronominal anaphors and proposed a workable solution in terms of an algorithm. A Sanskrit Analysis System (SAS) containing several modules has also been developed that will supposedly assist the resolution. We have modified their generalisations and have presented the algorithm based on our study of the language in the text. Thus, we have deviated from their proposal for reflexives and reciprocals.

Pralayankar et al. (2010) have presented an algorithm, which reportedly identifies different types of pronominals and their antecedents in Sanskrit. The computational grammar implemented there "uses very familiar concepts such as clause, subject, object etc., which are identified with the help of morphological information and concepts such as precede and follow". The method they have adopted for resolving the anaphors is by exploiting the morphological richness of the language. The system is reportedly giving encouraging results when tested with a small corpus. This work is also not flawless. In their algorithms they have ignored the fact that Sanskrit reflexives are impersonal, i.e. they do not encode person feature and also that Sanskrit reflexives do not agree in gender with their antecedents (see 2.2.1.). The algorithm does not consider reciprocals and cataphoric usage of reflexives and reciprocals. The paper does not define the term 'possessive' used in the algorithm.

1.3.5. The SARS Approach

The present approach, that is, *Sanskrit Anaphora Resolution System* (SARS) approach exploits the grammatical features of Sanskrit language. The Indian languages including Sanskrit are morphologically very rich and this richness is very useful in handling this task. The approach employed here uses mainly the nominal morphology. The input to the system is the POS tagged Sanskrit text in which various morphological features are attached to the words. The morphological features include features like case, number, gender, person, tense etc. Here we deal with intra-sentential anaphors. Thus, on the basis

14

of hierarchical POS tagging of the texts we train the system to recognize the anaphors and their antecedents.

1.4. Why Anaphora Resolution for Sanskrit?

As I have stated it in the beginning, Sanskrit is a language in which our ancient knowledge of various disciplines is stored. To use one's pre-existing knowledge for the betterment of humanity is always praiseworthy. However, doing so one needs to be very careful in deciding what is good and what is bad for the humanity with respect to time and circumstances in which one lives, and also, one should always keep them open to new research and developments in the field. A discipline is to open our eyes not to shut them down to new developments. So, as we have a huge literature of knowledge in this language, we want to share this knowledge with the rest of the world, and it is only possible when it is available in other world languages as well. The present work is such a small effort to help Sanskrit make translatable mechanically in other human languages so that it could be reachable to maximum number of people of the world. Anaphora resolution systems, as we know, are of crucial importance for correct translation. When translating into languages which mark the gender of pronoun, for example, it is essential to resolve the anaphoric relation. In translation, the reference to a discourse entity encoded by a source language anaphor by the speaker (or writer) has not only to be identified by the hearer (translator or translation system) but also re-encoded in a co-referential/co-specificational expression of a different language. So, to do the above things we need an anaphora resolution system for Sanskrit.

1.5. Methodology

The SARS approach of anaphora resolution takes advantage of the morphological richness of the Sanskrit language. First, the text was digitized in Unicode UTF-8 format (in fact I got it in digitized form from the Computational Sanskrit Research Lab, JNU and thanks to it), then the external (*anitya*) sandhis were split manually. The digitized text was proofread by me with the help of Shri Shyamacharan Pandey's recension (MLBD, 1975, reprint 2006). Afterwards, the entire text was manually tagged by me with the help

15

of annotation tool developed by MSRI. The tagging scheme (IL-POST) was also the one developed by MSRI, a hierarchical schema for tagging most of the Indian languages. As the text was in digital form, it was easy to find the various kinds of usage of pronominals and anaphors by searching them with the help of the MS word program. This method was very effective in ascertaining the frequency and availability of various forms. However, I have read the text along with its Hindi translation also to find different kinds of usage of pronouns and lexical anaphors. The sentences containing pronouns and lexical anaphors were picked up for examples and transcribed along with interlinear glosses and free translation in English. The pronouns and lexical anaphors have been classified according to their linguistic distribution. And finally, an algorithm was designed to handle the intra-sentential lexical anaphors and their antecedents automatically. To achieve this goal, methodologies of software development have been used. They include algorithm development, tools selection, data formatting, writing code of program, test and evaluation of the system. System is planned to develop in web architecture. The system tools are Java based. Front end is done in Java Server Pages (JSP) running on the web server called Apache Tomcat. The programming has been done in the Java environment. These things are given in detail in the Chapter 4.

1.5.1. The Data

The data in this work have been put uniformly. The Leipzig glossing rules have been followed for the annotation of the transcribed data. The transcription of the data has been done according to the transcription key given on page xiv. In the first line Sanskrit data have been given in italics using the GentiumAlt fonts, the second line is for the English gloss of the Sanskrit words and the third line is for free English translation of the given Sanskrit data. In the third line the source of the data has also been given in <angle brackets>. The source has to be read as first the name of the story from where the data are taken and then, the name of the section (*tantra*) of the text, given in acronyms, whence the story is taken.

16

1.6. Organization of the Chapters

This dissertation is divided in to five chapters. The first chapter, i.e. the current one, deals with the introductory things to the problem ahead and gives basic information about Sanskrit, PT, anaphora resolution, the various approaches to deal with this problem, and the present approach to handle this problem.

The second chapter "Pronouns and Lexical Anaphors in *Pañcatantra*" is a linguistic description of Sanskrit pronouns and anaphors available in the text. Their various kinds of forms and usage have been given. Pronouns and lexical anaphors have been classified and their compounding has also been discussed as per the given text. All this calisthenics helps in designing the algorithm for anaphora resolution.

The third chapter "SARS Design: Issues and Challenges" talks about the difficulties in resolving the anaphor and antecedent pairing. Sanskrit, as we know, is a relatively free word order language, and it is only the inflectional material used in the language which serves as a reliance to give the indication of the role of a particular word, in whatever place it has been used in a given sentence. Pronouns are often compounded with nominals, and the resolution of such pronouns is another kind of problem. This will require first the splitting of compounds (which is very complex process) and then the POS tagging of the segmented components of the compounds. However, we have not handled these cases in this dissertation. Sanskrit uses the device of cataphora also, and this has also to be taken care of. The irregularity of punctuation marks is a unique problem of Sanskrit which is unfavourable for anaphora resolution. Moreover, the stylistic of writing Sanskrit texts, especially the use of double *daṇḍa* in verses and in headings is not good for the tasks such as tokenization and sentence identification. At the end of the chapter an algorithm is proposed to resolve anaphors and cataphors mechanically.

"SARS Implementation" is the fourth chapter of this dissertation dealing with the designing of the system called SARS. It gives the detail of the implementation of the

17

system and the programming. The system architecture and design is explained with the sample code of JSP and Java. Finally, the implemented system's functioning is given with development information and screenshots.

In the fifth chapter "Conclusion and Further Enhancement" is the conclusion part which takes on all the work done through the dissertation and gives conclusive remarks on them. It gives information about the potential application of the system, the limitation of the system and further research to enhance the capacity of the system.

Chapter 2

Pronouns and Lexical Anaphors in *Pañcatantra*

The word pronoun is made up of two morphemes, viz. pro (meaning 'on behalf of') and noun: hence, a pronoun is traditionally said to be a word used in place of a noun expression. Pronouns differ from nouns in that they have no intrinsic descriptive content, and so are functors. This chapter discusses the usage of pronouns and lexical anaphors in the Sanskrit text *Pañcatantra*. All the data analyzed herein is taken from this text only. Many pronouns of Sanskrit have the ability to both stand alone as head of a noun phrase and to modify another noun phrase head. These properties will be discussed in passing throughout the chapter.

2.1. Personal pronouns

Sanskrit is a pronominally very rich language, and this is evident in the text of PT. All the personal pronouns encode the person and number of a referent. Persons are *first* (speaker), *second* (addressee) or *third* (person other than speaker or addressee, whether male or female). However, *bhavān* (masculine) and *bhavatī* (feminine) are addressees but are counted as third person and consequently, they take third person verb forms. As the present work is bound within the ambit of PT, I would be talking about pronouns available in this text only. The first and second person pronouns which have no distinction of gender have been sparsely used in the text, whereas the third person pronouns are available densely. The third person pronoun forms are also the forms of the demonstrative determiners. Like in Hindi/Urdu (Davison, 2000), they are part of a class of determiner pronouns which include relatives, interrogatives, and indefinites. All such pro-forms take syntactic antecedents. Their deictic usages are also possible. The third person pronouns encode the category of gender also along with person and number.

2.1.1. First Person Pronouns

In Sanskrit first person pronouns inflect for all the cases except vocative. The following table contains all the available forms used in the text. The alternative forms of the acc., dat., and gen. cases are found in first and second person in the language. However, in the text only dative singular (*me*), genitive singular (*me*) and genitive plural (*naḥ*) are available for first person. The alternative forms are found in second person pronoun as well.

Case/vibhakti	Singular	Dual	Plural
Nominative	*aham*	*āvām*	*vayam*
Accusative	*mām*	x[1]	*asmān*
Instrumental	*mayā*	x	*asmābhiḥ*
Dative	*mahyam, me*	x	x
Ablative	x	x	x
Genitive	*mama/ me, asmadīya/madīya*	*āvayoḥ*	*asmākam, naḥ*
Locative	*mayi*	x	x
Vocative	x	x	x

Table 4: First Person Pronouns available in the text

As is clear from the table, the use of dual forms of first person pronouns is very thin. Sanskrit pronominal morphology, as has been said earlier, is very rich. From the bases of the three personal pronouns, possessive adjectives are formed by means of *īya*. Such an adjective has the role of a genitive in the sentence and could be said the substitute of genitive forms in all the three personal pronouns. The first person possessive pronouns are *asmadīya* and *madīya,* which differ with respect to the number of the speakers. When speaker wants to refer himself only *madīya* form is used and when the speakers are more than one *asmadīya* is used. They agree with the case, gender and number features of the possessed noun~pronoun, as the adjectives do in Sanskrit. For instance,

---

[1] x = no form available for the slot in *PT*. However, in language they are available. This symbol has been used throughout this chapter for the same meaning.

(1)   *ayam*   **_asmadīya-ḥ_**   *bāndʰava-ḥ.*

        DEM.PRX.M.SG.NOM   1POSS.PL-M.SG.NOM   brother-M.SG.NOM

        'This is our brother.' <mūrkʰapaṇḍita-katʰā, apkr>

Herein, the demonstrative *ayam* and *asmadīya* are agreeing with the possessed noun *bāndʰavaḥ* with its case, gender and number features.

(2)   *yat*   **_asmadīya-m_**   *na* *hi* *tat*   *par-eṣām.*

        REL.NEU.3SG   1POSS.PL-NEU.SG   not DEF that.NEU.SG other-MAS.3PL.GEN

        'What is ours cannot be others.' <vaṇikputra- katʰā, mtsp>

Herein, *asmadīya* is in neutral gender due to its agreement with the relative antecedent *yat*.

(3)   **_madīya-ḥ_**   *namaskāra-ḥ* *vācya-ḥ*   *bʰagavat-aḥ.*

        1POSS.SG-M.SG greeting-M.SG communicable-M.SG lord-M.SG.GEN

        'Convey my greeting to Lord.' <caṭaka-kuñjara-katʰā, mtbd>

(4)   *yat*   *kāraṇam* *tena*   **_madīya-m_**   *yatʰā-avastʰita-m*

        REL.3SG.NEU reason   3SG.M.DST.INS 1POSS.SG-N.ACC as-present-N.ACC

        citta-m   *jñāta-m.*

        mind-N.ACC know.PSPL-N

        'For that reason, the state of my mind was understood by him.'

        <siṃhajambukaguhā- katʰā, kkly>

2.1.1.1. *First Person Pronoun Compounding*

Pronoun compounding is a unique feature of Sanskrit. All kinds of pronouns undergo such word formations. In the first person compounding the roots *asmat* (used for to refer more than one), and *mat* (used for singular) do participate and are the first components of such compounds. In the process of compounding the final sound /t/ of these roots is

TH 2294\

assimilated to the following sound, thus creating an instance of regressive voicing assimilation. Interestingly, these forms convey the meaning of possessive adjective only.

(5) *adya **asmat-svāmī** piṅgalaka-ḥ   bʰīta-ḥ   bʰīta-parivāra-ḥ   ca*
    today 1POSS.PL-master Pingalaka-NOM scared-NOM scared-family-NOM and
    vartate
    is
    'Today our master Pingalaka is scared and the same for his family.'
    <kīlotpāṭivānara-katʰā, mtbd>

(6) *yat   ete   **asmat-śatravaḥ**   an-upalabdʰa-asmat-vṛttānta-āḥ yatʰā*
    REL.N these 1POSS.PL-enemies not-got-1POSS-story-M.PL   as
    āga-tam   eva   yā-nti,   tataḥ mayā   na   kiṃcit
    come-PSPL emph   go-PL.PRS then 1SG.INS not anything
    kṛ-tam   bʰava-ti
    do-PSPL.NEU be-3SG.PRS
    'If these my enemies without knowing me, go as they came, then nothing was done by me.' <pipīlikābʰujaṅgam-katʰā, kkly>

(7) *avivekaḥ **asmad-bʰūpateḥ**   yat   purīṣa-utsarga-m   ācar-an*
    ignorance 1POSS.PL-king.SG.GEN that stool-excretion-ACC conduct-PRPL
    cirbʰaṭī-bʰkṣaṇam   karoti.
    cucumber-eating   do-3SG.PRS
    'It is our king's ignorance that he eats cucumber while toileting.'
    <dantilagorambʰayoḥ katʰā, mtbd>

(8) *tat   ekāntike **mad-vijñāp-yam**   avadʰārayantu devapādāḥ*
    then alone   1POSS.SG-apprise-worthy listen   sir
    'Then sir, please listen to my message alone.' <kīlotpāṭivānara-katʰā, mtbd>

(9)　　atra　ca　**mad-da-ttām**　　　　vṛttim　　　　bʰuŋjān-ānām　　paṇḍit-ānām

here　and　1POSS.SG-give-PSPL.F.ACC　stipend.F.ACC　consumer-PL.GEN　pundit-PL.GEN

paŋca-śatī　　tiṣṭʰa-ti.

five-hundred　sit-SG.PRS

'There are five hundred pundits present who consume stipend given by me.'

<katʰāmukʰam, mtbd>

(10)　sannihitaḥ　ca　ayam　pāpātmā　**mat-pati-ḥ.**

present　　and　this　wicked　1POSS.SG-husband-NOM

'And this wicked husband of mine is present here only.'　<dūtijambūkāṣādʰabʰūti-
katʰā, mtbd>

(11)　na　ca　tvayā　**mad-artʰam**　kṛpā　kāryā

not　and　you.INS　1POSS.SG-for　mercy　do.GRN.F

'There shouldn't be any mercy by you for me.'　<pipīlikābʰujaṅgam-katʰā, kkly>

There is a lack of such compounding for the possessive adjective *asmadīya* in the text.
However, in other Sanskrit texts there are occurrences where it is compounded with the
possessed noun and its case, number and gender features are gone. Thus, its root form is
allowed to compound with the possessed items and it remains the first component of such
construction, like others in this list. The form *madīya* which is a singular variant of
asmadīya has been used in the text.

(12)　**madīya-bʰār-eṇa**　　　ati-śrānta-ḥ　　tvam

1POSS.SG-burden-M.SG.INS　very-tired-M.SG　you.NOM

'You are very tired with my burden.'　<bakakarkaṭaka-katʰā, mtbd>

2.1.2. Second Person Pronouns

Like first person pronouns, second person pronouns also inflect for all cases except vocative. Morphologically, they are similar to first person pronouns. Like in the first person, the alternative forms of the accusative, dative, and genitive cases are found in second person as well in the language. However, in the text the alternative forms have been used in accusative plural (*vah*) and genitive singular (*te*) only. The following table contains all the second person pronoun forms used in the text:

Case/vibhakti	Singular	Dual	Plural
Nominative	*tvam*	*yuvām*	*yūyam*
Accusative	*tvām*	x	*vah*
Instrumental	*tvyā*	*yuvābhyām*	*yuṣmābʰih*
Dative	*tubhyam*	x	x
Ablative	x	x	x
Genitive	*tava, te, tvadīya, tvadīyasya, yuṣmadīyam*	x	*yuṣmākam*
Locative	*tvayi*	x	*x*
Vocative	x	x	x

Table 5: Second Person Pronouns available in the text

The second person possessive pronouns are *yuṣmadīya* and *tvadīya* which are derived from the roots *yuṣmad* and *tvad* respectively by means of *īya*. These possessive adjectives agree with the case, gender and number features of the possessed noun, as the first person possessive adjectives do. However, when *yuṣmadīya* and *tvadīya* are compounded with the possessed noun, their number, gender and case features are lost. The form *yuṣmadīya* has been used for plural and *tvadīya* for singular in the text.

(13)   *param vayam vana-carāh*   ***yuṣmadīya-m*** *ca   jalānte*   ***gṛha-m.***

       but   we   forest-dweller.M.PL 2POSS.PL-N.SG and water-inside home-N.SG

       'But, we stay in forest and your abode is in the water.' <prastāvanā-katʰā, ldpr>

(14)   *tat   rātrau   api*   ***tvadīya-m***   *matʰa-m*   *tyaktvā*

       then night\SG.LOC EMPH 2POSS.SG-M.SG.ACC monastery-M.SG.ACC leave.PSPL

anyatra   *mat^he*   *yā-syāmi.*

else_where   monastery\SG.LOC   go-1SG.PRS

'Even in night leaving your monastery, I will go to another one.'

<hiraṇyakatāmracūḍa-kat^hā, mtsp>

(15)   *na asti sā*   <u>*tvadīy-ā*</u>   *tulā*

not is   3SG.DEF.F   2POSS.SG-F.SG.NOM   scale.F.SG.NOM

'That scale of yours is no more.' <*lohatulāvaṇikpura-kat^hā, mtbd*>

2.1.2.1. Second Person Pronoun Compounding

Like first person pronouns, the roots of second person pronoun also undergo for compounding. The root forms *yuṣmad* and *tvad* are used in compound constructions for plural and singular respectively. Interestingly, these forms convey the meaning of possessive adjective. Like in the first person pronoun compounding, the voicing assimilation takes place here also.

(16)   ***yuṣmad-darśana-mātra-anurakta-yā***   *mayā*   *ātmā*   *prada-ttaḥ*   *ayam.*

2POSS.PL-appearance-only-infatuated-F.INS   1SG.INS   self   give-PSPL.M   this

'I got fascinated by seeing you only and have given myself to you.' <vaṇikputra-kat^hā, mtsp>

(17)   ***tvad-varjam***   *anyaḥ b^hartā*   *manas-i*   *api*   *me*   *na*   *b^hav-iṣyati*   *iti*

2POSS.SG-barring   other   husband   mind-SG.LOC   EMPH   my   not   be-3SG.FUT   QUOT

'I can't have another husband even in my mind except you.' <vaṇikputra-kat^hā, mtsp>

(18)   *kintu*   ***tvat-prārt^hanā-siddhy-art^hyam***   *sarasvati-vinodam*   *kar-iṣyāmi*

but   2POSS.SG-prayer-completion-for   Saraswati-entertainment   do-1SG.FUT

25

'But, for the completion of your prayer, I will be entertained with goddess Saraswati.' <kat^hāmuk^ham, mtbd>

(19) *aho gajaḥ   ayam yuṣmat-kula-śatruḥ*
oh! elephant this   2POSS.PL-family-enemy
'Oh! this elephant is your family enemy.' <siṃhaśṛgālaputrayoḥ kat^hā, ldpr>

(20) *saḥ prāha - "deva! saṃjīvaka   yuṣmat-pād-ānām upari drohabuddʰi iti....."*
he said – " Sir! Sanjeevaka 2POSS.PL-foot-PL.GEN on   against   QUOT
'He said- "Sir! Sanjeevak is against you......".' <siṃhaśaśaka-kat^hā, mtbd>

The second person possessive pronouns *yuṣmadīya* and *tvadīya* also make compounds in the language, but in the text only *tvadīya* has been used in such construction, where it has lost its grammatical features of case, number and gender, as was the case with first person possessive pronouns.

(21) *yadyapi tvadīya-vacanam   na karoti, tathāpi svami*
though 2POSS.SG-word-N.SG.ACC not does, still   master
sva-doṣa-nāś-āya   vācyaḥ.
self-fault-destuction-SG.DAT speakable
'Though the master does not pay any heed to your words, still for the destruction of your own faults you should speak to him.' <dantilagorambhayoḥ kat^hā, mtbd>

2.1.3. Third Person Pronouns, Definites, Indefinites, Relatives and Interrogatives

Third person pronouns have distinct forms for the categories of number, gender, and proximity, and they inflect for case. To express the honorific status of the referent their plural forms are used. They are also used as demonstratives and definites. They occur, sometimes, with determiners and pre-nominal adjectives as nouns do. The indefinite pronouns are formed by suffixing *cit* or *can* to the inflected forms of the interrogative pronouns, and after this suffixation they are considered *avyayas* (indeclinables) in the

26

traditional grammar. The definite and indefinite usage of pronouns is densely available in the text. The table below contains all such forms used in the text.

(*x = no form available for the slot in PT)

Case	Singular	Dual	Plural
Nominative	*saḥ* (3SG.M.DST), *tat* (3SG.N.DST), *asau* (this.M), *idam* (this.PRX.N), *ayam* (this.M), *iyam* (this.PRX.F), *etat* (this.N), *bhavān* (you.3SG.HON), *anyaḥ* (othr.M), *ekaḥ* (one.M), *aparam* (other.N), *aśeṣa* (all), *yaḥ* (REL.M), *yat* (REL.N), *yā* (REL.F), *kaḥ* (who.M), *kim* (what.N), *ko'pi* (some_body.M), *kiṃcit* (some.N), *kīdṛśī* (how.F)	*tau* (3DU.M. DST)	*te* (3PL.M.DST), *ete* (3PL.M.PRX), *ye* (REL.PL), *sarve* (all.3PL.M)
Accusative	*svam* (self.M), *tam* (3SG.M.DST), *etad* (3SG.N.PRX), *yam* (REL.M), *yat* (REL.N), *kam* (which.3SG), *enam* (3SG.M.PRX)	x	*etān* (3PL.M.PRX), *tān* (3PL.M.DST), *sarvān* (all.3PL.M)
Instrumental	*tena* (3SG.M.DST), *tayā* (3SG.F.DST), *yena* (REL.3SG.M), *anyena* (other.3SG.M), *kena* (who.3SG.M), *anena* (3SG.M.PRX), *ātmanā* (self.3SG), *ekatamena* (one.3SG.M), *taddvāreṇ* (3SG.M.DST)	x	*sarvaiḥ* (all.3PL.M), *bhavadbhiḥ* (you.3PL.M)
Dative	*tasmai* (3SG.M.DST)	x	x
Ablative	*etasmāt* (3SG.M.PRX), *tasmāt* (3SG.M.DST), *kasmāt* (who.3SG.M)	x	x
Genitive	*tasya* (3SG.M.DST), *asya* (3SG.M.PRX), *yasya* (REL.3SG.M), *ekasya* (one.3SG.M), *kasya* (what.3SG.M) *kasyacit* (INDEF.M) *bhavataḥ* (you.3SG.M)	*tayoḥ* (3DU.M. DST)	*eteṣām* (3PL.M.PRX), *bhavatām* (you.3PL.M), *yeṣām* (REL.3PL.M), *sarveṣām* (all.3PL.M), *teṣām* (3PL.M.DST), *ekeṣām* (one.3PL.M), *anyeṣām* (other.3PL.M)
Locative	*tasmin* (3SG.M.DST), *yasmin* (REL.3SG.M), *asmin* (3SG.M.PRX), *etasmin* (3SG.M.PRX), *kasminścit* (INDEF.3SG)	x	x

Table 6: Third Person Pronouns, Relatives, Definites, Indefinites and Interrogatives in the text

The third person pronouns are also used as determiners. They are part of a class of determiner pronouns which includes relatives, interrogatives, demonstratives, definites

27

and indefinites. All of them take syntactic antecedents. Deictic usages are also possible. Apart from this, like Hindi, a lot of adverbial pronouns are also found in the text, such as *atra* 'here', *tatra* 'there', *itaḥ* 'from here', *anyatra* 'some where else', *iha* 'here'. The third person possessive pronoun *tadīya* has been used in compounds only in the text (see 2.1.3.5).

2.1.3.1. Definites

In the example below the third person pronoun *saḥ* has been used as definite determiner before second person genitive.

(22) **saḥ**   *te*   *mātula-ḥ*   *api*   *na*   *āyataḥ?*

     3SG.DEF.M.NOM   2SG.GEN   maternal_uncle-M.NOM   EMPH   not   came

     'Even your uncle didn't come?' <bakakarkaṭaka-katʰā, mtbd>

In the following sentence the root of the distal third person pronoun *tad* has been compounded with the proximal third person pronoun form *etān*. In its stand alone form it would have been *tān*. It serves here merely the purpose of definiteness.

(23) **tad-** *etān*   *paśyataḥ*   *me*   *mahat api*   *rājyam*   *na saukʰyam āvahati.*

     DEF- 3PL.M.PRX   see.prpl.m   1SG.DAT   huge   EMPH   kingdom   not   pleasure   brings

     'Seeing these (children), even this huge kindom does not bring me any

     happiness.' <katʰāmukʰam, mtbd>

2.1.3.2. Indefinites

In Sanskrit, as I have stated earlier, the interrogative pronouns suffixed with *cit* or *can* make indefinites. In the following sentence *kaścit* is a compound of *kaḥ* (who.M.SG.NOM) and *cit*. This *kaścit* has been used to denote the indefiniteness of a proper noun, namely, Gomayu, a jackal.

(24) **kaścit** *gomāyuḥ*   *nāma*   *śṛgālaḥ*   *kṣutkṣāmakaṇtʰaḥ itaḥ*   *tataḥ*

     INDEF   Gomayu.M.NOM   named   jackal   hungry   here   there

parib^hraman vane   sainya-dwaya-saṃgrāma-b^hūmi-m   apaśyat.
wandering forest.LOC army-couple-battle-field.F-ACC   see\PST.3SG
'A hungry jackal named Gomayu wandering hither and thither saw a battle field
of two armies in a forest.' <śṛgāladundub^hi-kat^hā, mtbd>

In the following example *kaścit* has been used as an indefinite determiner to express the
indefiniteness of the village in the sentence.

(25) *at^ha   evam gacc^han   sāyaṃtana-samay-e kaścit grāma-m   āsasāda.*
afterwards thus wandering evening-time-SG.LOC INDEF village-ACC got
'In this way, wandering in the evening, he got some village.'
<dūtijambūkāṣād^hab^hūti-kat^hā, mtbd>

The following is an instance of instrumental use of the indefinite pronoun.

(26) *tataḥ   ca aham   āgacc^han antarāle   mahat-ā   kenacit*
afterwards and 1SG.NOM coming   meanwhile huge-SG.INS INDEF.SG.INS
apar-eṇa   siṃh-ena   kṣiti-vivar-āt   nirgatya ab^hihitaḥ......
other-SG.INS lion-SG.M.INS earth-hole-SG.ABL came out tell.PSPL.SG.M
'After this, while coming, I was told by some other big lion that came out of a
hole of the earth.............' <siṃhaśaśaka-kat^hā, mtbd>

The indefinite determiner could be preceded by the noun of the determiner phrase. In the
next sentence the indefinite determiner *kiñcit* has been preceded by the noun *śāstram*:

(27) *tat saṃkṣepamātram śāstra-m   kiñcit   eteṣām*
then brief-only   discipline-SG.N.ACC INDEF.SG.ACC 3PL.PRX.GEN
prabod^hana-art^ham cintyatām iti.
awakening-for   think_of QUOT
'Then, think of some brief discipline for awakening of these children.'
<kat^hāmuk^ham, mtbd>

29

2.1.3.3. Relative Pronouns

Following are some of the examples of relative pronouns:

(28) **yaḥ**   *na ātmanā na ca pareṇa   ca bandʰuvarg-e*

     REL.3SG.M.NOM not self.INS not and other.SG.M.INS and brethren.LOC

     dīne   dayām   na kurute na ca bʰṛtya-varg-e.

     poor.LOC compassion.F.ACC not does not and servant-group-LOC

     kim tasya   jīvita-pʰalam hi manuṣya-lok-e

     what 3SG.DST.GEN life-fruit   EXPL human-world-LOC

     kākaḥ   api jīvati ciram ca bali-m   ca bʰuṅkte.

     crow.M.NOM also lives long and offering-ACC and eats

     'What is the usefulness of the life of the person for this world who does not help

     by his own or by others to his fellow people, the poor and servants? A crow also

     lives long and eats offerings.' < kīlotpāṭi-vānarakatʰā, mtbd>

(29) **ye**   *ca prāhuḥ duḥ-ātmānaḥ duḥ-ārādʰyāḥ   mahībʰujaḥ*

     REL.3PL.M.NOM and say.PL.PRS bad-souls   difficult-worshipable.PL kings

     pramāda-ālasya-jāḍyāni   kʰyāpitāni   nijāni   taiḥ

     carelessness-laziness-foolishness.N.PL report.PSPL.N.PL self.N.PL 3PL.M.DST.INS

     'The people who say kings are wicked, and difficult to be pleased; carelessness,

     laziness, and foolishness are advertised by them only.' <kīlotpāṭi-vānarakatʰā,

     mtbd>

2.1.3.4. Interrogative Pronouns

In the text *kim, kaḥ, and kasya* interrogative pronouns have been used.

(30) **kim**   *tena   jātu jātena   mātuḥ*

     what.SG.N 3SG.M.DST.INS EXPL production.INS mother.SG.GEN

     yauvana-hāri-ṇā?   ārohati   na yaḥ   svasya

     youth-depriver-SG.M.INS climb.3SG.PRS not REL.3SG.NOM self.SG.M.GEN

vaṃśa-sya   *agre*   *dʰvajaḥ yatʰā.*

lineage-SG.M.GEN before flag as

'What is the use of the birth of a son, the destroyer of his mother's youth, who
does not rise up to his family like a flag?' <kīlotpāṭi-vānarakatʰā, mtbd>

(31)   *kaḥ*   *artʰān*   *prāpya na garvitaḥ viṣayiṇaḥ*

who.M.SG.NOM wealth.PL.M.ACC get.pspl not proud sensualist.GEN

kasya   *āpadaḥ*   *astam*   *gatāḥ.*

who.SG.M.GEN calamities end.ACC go.PSPL.PL

strī-bʰiḥ   *ka-sya*   *na kʰaṇḍitam bʰuvi*   *manaḥ* **kaḥ**   *nāma*

woman-PL.INS who-M.SG.GEN not broken world.LOC heart who.M.SG.NOM EXPL

rājñām   *priyaḥ.*

king.PL.GEN dear

kaḥ   *kāla-sya*   *na gocara-antaragata-ḥ*   **kaḥ**   *artʰī*   *gataḥ*

who.SG.NOM time.SG.GEN not visible-subjected-SG.M who.SG beggar go.PSPL-SG

gauravam

glory-ACC

kaḥ   *vā durjana-vāgurā-su*   *patitaḥ kṣemeṇa*   *yātaḥ*   *pumān.*

who.SG.NOM or wicked-coaxing-PL.LOC fallen peacefully go.PSPL.M man

'Who does not become proud after getting wealth? Which sensualist's calamities
are over? In this world whose heart was not broken by women? Who is dear to
kings? Who is not visible to time? Which beggar did acquire glory? Which man
did pass through peacefully the coaxing of wicked people?' <dantilagorambʰayoḥ
katʰā, mtbd>

2.1.3.5. *Third Person Pronoun Compounding*

The compounding of third person possessive pronoun with nominals is available in the
text. The root of the third person pronoun has not been used in the compound
constructions in the text. The base forms of the possessive adjectives *tadīya* and
bʰavadīya have been used in compound constructions as was the case with first and
second person possessive adjectives. As I have said earlier, the stand alone usages of

tadīya are not available in the text, and the same case is with *bʰavadīya*. Here are some examples:

(32)   *matta-ibʰa-kumbʰa-pariṇāhini kuṅkum-ārdr-e tasyāḥ payodʰara-yug-e*
       wild-elephant-cheek-having crocus-wet-LOC her   breast-dual-LOC

       rati-kʰeda-kʰinnaḥ.
       sex-languor-exausted

       vakṣaḥ nidʰāya bʰuja-paṅjara-madʰyavartī svapsye   kadā kṣaṇam
       chest put.PSPL arm-skeleton-between   sleep.1SG.FUT when moment

       avāpya   tadīya-saṅgam.
       gain.GRN 3POSS.SG-company

       'When will I, exhausted from sex, sleep in her lap putting my chest on her breasts which are like wild elephant's cheek and wet from the liquid of crocus?'
       <viṣṇurūpadʰṛkkaulika-katʰā, mtbd>

(33)   *api ca ayam taiḥ   viprakṛtaḥ asmākam eva   pusṭa-ye*
       EXPL and this 3SG.DST.PL. INS offended 1PL:GEN   EMPH growth-DAT

       bʰaviṣyati, tadīya-randʰra-darśan-āya ca, iti   aneka-kāraṇ-ena   ayam
       be.3SG.FUT 3POSS.SG-defect-show-SG.DAT and QUOT many-reason-SG.INS this

       a-vadʰyaḥ   iti
       not-killable QUOT

       'And this offended by them would be useful for us only as he will tell us their weaknesses. For many such reasons this is not killable.' <cauravṛdʰavaṇik-katʰā, kkly>

(34)   **bʰavadīya-sāhas-ena**   *aham   tuṣṭaḥ.*
       3POSS.SG-courage-SG.INS 1SG.NOM satisfied

       'I am satisfied with your courage.' <mandabʰāgyasomilaka-katʰā, mtsp>

2.1.4. Conjunctions Homophonous with Pronouns/Demonstratives

Many pronominal forms are homophonous with conjunctions. Among them *tasmāt, tat, yat,* and *yena* have been mainly used. These words serve as linkers and they join the previous sentence to the current sentence. Their word order is also fixed as they occur at the end of the preceding sentence and beginning of the following sentence. As the table of third person pronouns shows, these forms are used to refer to some person(s) also. Here, I will give some examples to show how they have been used.

(35) yā   punaḥ tri-stanī   kanyā   yā-ti   locana-gocaram.
      REL.F   again three-breast_with   daughter go-3SG.PRS eye-visible

      pitara-m   nāśa-yati   eva   sā   drutam, na   atra saṃsayaḥ.
      father.ACC destroy-3SG.PRS EMPH   she   quickly not here doubt

      tasmāt asyāḥ   darśanam   pariharatu   devaḥ.
      SO   3SG.PROX.F.GEN appearance remove.IMP.3SG lord

      'And, the daughter with three breasts when passes through the eyes of her father, definitely, destroys him quickly and there is no doubt in it. So, Lord, remove its appearance.' <andʰakakubjakatristanī-katʰā, apkr>

Herein *tasmat* is functioning like a linker and links the preceding verse with the following sentence:

(36) kaulikaḥ āha- "bʰoḥ! kim aham karomi   dāru-sāmagrī-m   vinā   me
      Kaulika said   hey! what I   do.1SG woods-apparatus-ACC without my

      kuṭumbam bubʰukṣayā pīḍyate **tasmāt** anyatra   śīgʰram gamyatām."
      family   hunger.INS suffers so   elsewhere soon   go

      'Kaulika said - " Hey, what to do, without woods my family suffers from hunger, so please go somewhere else hurriedly".' <mantʰarakaulika-katʰā, apkr>

In the following sentence *tasmāt* and *yena* both has been used to join sentences.

33

(37) *mat-pṛṣtʰa-gataḥ   tila-mātram   api   cali-tum na   śakno-ti*
1POSS.SG-back-go.PSPL sesame-quantity EMPH go-GRN not can-3SG.PRS

tasmāt katʰa-yāmi asya   nija-abʰiprāyam   yena   abʰīṣta-devatā-
so · say-1SG.PRS 3SG.PRX.GEN self-intention.ACC so_that dear-god-

smaraṇam   karoti.
remembrance does

'(this monkey) seated on my back cannot move even for an inch, so, I say my intention to him, so that he prays his desired god.' <prastāvanā-katʰā, ldpr>

(38) *tat   enam   māyā-vacan-aiḥ   viśvās-ya   aham   cʰātra-tām*
then 3SG.PRX.ACC false-word.PL.INS convince-GRN 1SG.NOM student-ness.ACC

vrajāmi   yena   saḥ   viśvas-taḥ   kadācit   mama   hastagataḥ bʰaviṣyati.
go.1SG.PRS so_that he convince-PSPL perhaps 1SG.GEN hand-came be.FUT.3SG

'Then, convincing him by false words, I become his disciple, so that he would be convinced, perhaps, will be in my hands.' <dūtijambūkāṣādʰabʰūti-katʰā, mtbd>

(39) *tat   yatʰā anuśrūyate   asti dākṣiṇātye   janapade   mahilāropyam nām*
then as · hear.PRS.PASS is southern.LOC district.LOC Mahilaropya named

nagaram.
city

'Then, as it is heard, there is a city named Mahilaropya in southern district.'
<katʰāmukʰam, mtbd>

(40) *tat   saṃkṣepamātram śāstram   kiñcit eteṣām   prabodʰana-artʰam cintyatām*
then brief-only discipline some 3SG.PL.GEN awakening-for think

iti.
QUOT

'Then, think of some brief discipline for awakening these children.'
<katʰāmukʰam, mtbd>

(41) *yat prabʰūte api vitte artʰopāyāḥ cintanīyāḥ kartavyāḥ ca iti*

that enormous EMPH wealth.LOC wealth-means thinkable.PL doable.PL and QUOT

'.....that, even after having a lot of money, the solutions for earning money is

thinkable and doable.' <prastāvanākatʰā, mtbd>

2.2. Lexical Anaphors

The Sanskrit language is enormously rich in its vocabulary and other linguistic devices
that make this language highly suitable for any kind of communicative purpose. In terms
of lexical anaphors also it is rich. Though some of the Indo-Aryan languages, like
Marathi, Gujarati and Sinhala, have verbal anaphors also along with nominal ones
(Subbarao, ms.), Sanskrit has only the nominal form of the anaphors: reflexives and
reciprocals. The reflexive pronouns, like any nominal, inflect for case (barring *svayam*).
The possessive reflexives, which could be more appropriately baptized as 'reflexive
possessive adjectives', even inflect for number and gender also. The reciprocals,
however, are stagnant forms and do not inflect for any grammatical feature. The
reflexives and reciprocals are also used in compounds.

2.2.1. Reflexives

The reflexives found in the text are invariant for person feature, in other words, Sanskrit
reflexives do not encode person feature. They are found invariably in all persons. The
reflexive pronouns found in PT could be classified in four categories:

2.2.1.1. Bare Reflexives

In the text I found *ātman* as the only proper reflexive pronoun. Like personal pronouns, it
takes case suffixes, but it is invariant for gender and person features. In contrast to
personal pronouns, referent *number* is not a coded feature of the reflexive *ātman* (barring
its genitive forms). However, its genitive forms behave like possessive reflexives and
agree with the number of the antecedent (see under possessive reflexives), but unlike
other possessive reflexives, they do not vary for gender agreement with the possessed

35

item. The following table contains all the morpho-syntactic variants of *ātman* available in the text.

Case/vibhakti	Singular	Dual	Plural
Nominative	*ātmā*	x	x
Accusative	*ātmānam*	x	x
Instrumental	*ātmanā*	x	x
Dative	x	x	x
Ablative	x	x	x
Genitive	*ātmanah**	x	*ātmanām**
Locative	*ātmani*	x	x
Vocative	x	x	x

Table 7: Casal inflections of *ātman* used in the text

* The genitive inflections of *atman* are naturally used as possessive reflexives, but the difference between these forms and other possessive reflexive is that the genitive variants of *atman* do not encode gender feature, while others do so. The examples are given in the section 2.2.1.3.6.

In Sanskrit the reflexives and the modifiers of their antecedent can occur adjacent to each other and they agree with each other. This is exemplified in the sentence below where *ātmānam* is agreeing with the modifier *aratnam*. This is unique feature of Sanskrit reflexives. (Here and throughout the chapter anaphors/cataphors and their antecedents are indicated by boldface, and also if there is an agreement between anaphor and some other element, it is denoted by underline.)

(42) *tataḥ* **aham** *pūrvam eva* **ātmāna-m** *aratna-m* *samarpya*

then 1SG.NOM before EMPH self-ACC jewellery_less-ACC surrender.GRN

etān *muñcāmi.*

these.M.3PL.ACC release.1SG.PRS

'Then, I being without any jewellary, hereby, dedicate myself and get these people released.' <caurabrāhmṇa-kathā, mtbd>

The following sentence is an example of reflexive used in singular number to refer multiple antecedents. Speijer (1886) also observes the same, "It *(atman)* is always a masculine and a singular, even when referring to a plural or a not-masculine."

(43) *atha   tasmin   vyādʰ-e   dūrataram gat-e   sarve api   te*

afterwards 3SG.M.LOC hunter-LOC farther   go.PSPL-LOC all   EMPH those

kāka-kūrma-mṛga- mūṣak-āḥ *param-ānanda-bʰājaḥ   parasparam*

crow- turtle-deer-rat-M.PL   superior-pleasure-receiver.M.PL REC

*āliṅg-ya   punaḥ jātam iva   **ātmānam** manyamānāḥ tat-eva   saraḥ*

hug-GRN   again born as_if REF.SG.ACC assuming.PL   that-EMPH pond

samprāp-ya mahā-sukʰ-ena   subʰāṣita-kathā-goṣṭʰī-vinod-ena

obtain-GRN great-pleasure-SG.INS maxim-story-seminar-amusement-SG.INS

kālam   nay-anti   sma.

time.ACC   lead-PL.PRS PST

'Afetrwards, the hunter being gone far off, all those crows, turtles, deer, and rats embraced each other with pleasure and assuming themselves as resurrected, having reached to that pond, spent time by maxims, stories, and meetings with great pleasure.' <vṛṣabʰānugaśṛgāla-katʰā, mtsp>

In the sentence below the *ātman* reflexive has been used as an accusative case marked and as instrumental case marked separately in one sentence and they are co-referential.

(44) *na   śaknomi   **ātmānam** api   **ātmanā** vodʰum   kim punaḥ enam*

not can.1SG.PRS self.ACC   EMPH self.INS bear.GRN what again 3SG.PRX.ACC

paṅgum.

lame_person

'I cannot bear even myself, what to do with this lame person?'

<brāhmaṇadampatyoḥ katʰā, ldpr>

37

The *atman* reflexive could be coreferential with a relative pronoun also. In the example below *atman* in instrumental case is coreferential with *yaḥ* (REL.NOM).

(45) *yaḥ*   *na*   *ātmanā*   *na*   *ca*   *pareṇa*   *ca*   *bandʰu-varg-e*
REL.SG.M.NOM   not   self.INS   not   and   other.SG.INS   and   brother-group-LOC

dīn-e   *dayā-m*   *na*   *kurute*   *na*   *ca*   *bʰṛtya-varg-e*
poor-LOC   compassion-ACC   not   do.3SG.PASS.PRS   not   and   servant-group-LOC

kim   *tasya*   *jīvita-pʰalam*   *hi*   *manuṣya-lok-e*
what   3SG.M.GEN   life-fruit   EXPL   human-world-LOC

kāka-ḥ   *api*   *jīvati*   *ciram*   *ca*   *bali-m*   *ca*   *bʰuṅkte*
crow-SG.NOM   also   lives   long   and   offering-ACC   and   eat.3SG.PASS.PRS

'What is the usefulness of the life of the person for this world who does not help by himself or by others to his fellow people, the poor and servants? A crow also lives long and eats offerings.' <kīlotpāṭi-vānarakatʰā, mtbd>

Following is an example of locative use of *ātman*; it is to be noted that it is in singular form and co-referential with *āpaḥ* (water.PL).

(46) *duḥ-ārādʰy-āḥ*   *śriyaḥ*   *rajñ-ām*   *duḥ-āp-āḥ*
difficult-please-PL   wealth_goddess.PL   king-PL.GEN   difficult-achieve-PL

duḥ-parigrah-āḥ   *tiṣṭʰ-anti*   *āpaḥ*   *iva*   *ādʰār-e*   *ciram*
difficult-protect.PL   sit-3PL.PRS   water.PL   like   base.LOC   long_time

ātman-i   *saṃstʰitāḥ.*
self-SG.LOC   placed.PL

'The goddess of wealth of kings cannot be pleased, achieved and protected easily. She stands like water based in its own base.' <kīlotpāṭi-vānarakatʰā, mtbd>

The word *atman* is used in the meaning of soul also and inflects for cases:

(47) *yaḥ*   *karoti*   *nara-ḥ*   *pāpa-m*   *na*   *tasya*   *ātmā*   *dʰruvam*   *priyaḥ*

REL.SG.NOM does   man-SG.NOM sin-ACC not 3SG.M.GEN soul   DEF   dear

ātman-ā hi   kr̥-tam   pāpam ātman-ā eva   hi   bʰujyate

soul-INS DEF do-PSPL sin   soul-INS EMPH DEF eat.3SG.PASS.PRS

'This is firmed that the person who does sins, he does not love his own soul,

because the sins done by soul ~ self are experienced by soul ~ self only.'

<kapotalubdʰaka-katʰā, kkly>

(48)   *atʰavā ātmā sām-ādi-bʰih̥   upāy-aih̥   abʰiraksanīyah̥*

       or   soul kind_words_etc.-PL.INS means-PL.INS protectable

       'Or self ~ soul is protectable by means of kind words, etc.' < caṭaka-kuɲjara-
       katʰā, mtbd>

(49)   *yuṣmat-darśana-mātra-anurakta-yā   mayā   ātmā prada-ttah̥ ayam*

       2.POSS.PL-appearance-only-infatuated-SG.F.INS 1SG.INS self   give-PSPL this

       'I got fascinated by seeing you only and have given myself to you.' <vanikputra-
       katʰā, mtsp>

2.2.1.2. *Intensive Reflexives*

Unlike other reflexive pronouns, the intensive reflexive pronouns *svayam* and *svayameva*
are morphologically invariant and refer to the subject of the sentence. Intensive reflexives
are like emphatic markers, and that's why they are called emphatic reflexives as well. In
the text, two such reflexives are available: *svayam* and *svayam-eva*. The reflexive
svayam-eva is semantically more emphatic; *eva* itself is an emphatic in the language.
Otherwise *svayam* and *svayam-eva* are the same. Both of them do not occupy an
argument position in a sentence but serve merely as emphatic. In contrast to personal
pronouns, referent number is not a coded feature of Sanskrit intensive reflexive pronouns.
This type of reflexive too is used in compounds (see 2.2.1.4.6).

(50)   *viṣa-vr̥kṣah̥ api   samvardʰ-ya svayam cʰettum asāmpratam.*

poison-tree EMPH grow-GRN   self   cut.INF improper

'It is improper even to cut the poison tree by self which you have grown up.'

<siṃha-śaśaka-katʰā, mtbd>

(51) *bʰṛty-aiḥ*   *vinā*   ***svayam rājā*** *loka-anugraha-kāribʰiḥ*

      servant.PL.INS without self   king people-welfare-doer.PL.INS

      mayūkʰ-aiḥ iva dīptāṃśuḥ tejasvī   api   na   śobʰa-te

      ray-PL.INS   like sun   brilliant EMPH not suit-3SG.PRS.PASS

      'As without rays the brilliant sun does not look beautiful, so is a king himself
      without servants who serve people.' <kīlotpāṭi-vānarakatʰā, mtbd>

svayam could be used to modify a past participle also. In the following example *svayam*
is modifying the past participle *āgatam*:

(52) *muhūrtam api   an-ātmīyam bhok-tum na   labh-yate.*   ___svayam___

      moment   EMPH not-self's   eat.INF   not attain-3SG.PRS.PASS self

      ___āga-tam___ *api   vidʰi-nā   apahri-yate.*

      come-PSPL EMPH fate-SG.INS abduct-3SG.PRS.PASS

      'You can't have it even for a moment if it is not yours. If it has come to you itself,
      it will be taken away by fate.' <vaṇikputra-katʰā, mtsp>

From (53) to (56) are examples given for *svayam-eva* which is, as has been said already,
semantically more emphatic. In the language *eva* is used as emphatic marker. As *svayam*
is itself an emphatic marker, so, the addition of *eva* just intensifies its meaning. Thus *eva*
serves as an intensifier when compounded with *svayam*.

(53) ***saḥ***   *ca*   *tam*   *parijñāya samīpa-m gatvā* ***svayam-eva*** *kautukāt*

      3SG.NOM and 3SG.ACC know.GRN near-ACC go.GRN self-EMPH   curiosity.ABL

      atāḍayat.

      beat.3SG.PST

'After knowing well the drum he went nearer and himself beaten it up curiously.' <śṛgāla-dundubʰi-katʰā, mtbd>

(54)   *tat katʰam **svayam-eva** vyāpāda-yāmi?*
then how   self-EMPH   kill-1SG.PRS
'Then, how I kill?' <siṃha-śaśaka-katʰā, mtbd>

(55)   *prakaṭī-kṛtam **tvayā svayam-eva** ātmanaḥ duṣṭatvam kauṭilyam   ca.*
reveal-PSPL   2SG.INS self-EMPH   self.GEN   wickedness crookedness and
'Your wickedness and crookedness have been revealed by yourself only.'
<bakanakula-katʰā, mtbd>

(56)   *tat yadi **saḥ**   **svayam-eva** ātmānam vadʰ-āya   niyoja-yati tadā*
then if   3SG.NOM self-EMPH   self.ACC   killing-DAT put-3SG.PRS then
vadʰ-yaḥ.
kill-GRN
'Then, if he presents himself for killing then he should be killed.'
<uṣṭrakākādikatʰā, mtbd>

2.2.1.3. *Possessive Reflexives*

sva, svīya, svaka, svakīya, ātmīya, and *nija* are possessive reflexives and are inflected for case, number and gender features. All of these possessive reflexives are semantically identical and mutually interchangeable. They are used more like an adjective and generally precede the modified element. The genitive forms of *ātman* also behave like possessive reflexive but they do not agree with the possessed item in gender. Below, the table is accommodated with the possessive reflexive forms used in the text.

Case/vibhakti	Singular	Dual	Plural
Nominative	x	x	*nijāḥ*
Accusative	*svam, nijām, svakam, svakīyam, ātmīyam, anātmīyam*	x	*nijāni*
INSumental	x	x	*svaiḥ*
Dative	x	x	
Ablative	x	x	
Genitive	*svasya, ātmanaḥ*	x	*ātmanām*
Locative	x	x	*sveṣu*
Vocative	x	x	x

Table 8: Possessive reflexives available in the text

2.2.1.3.1. *sva*

Though *sva* itself is a possessive reflexive, but still inflects for genitive along with other cases, as is clear from the following examples:

(57) *kim   tena   jātu jātena   mātuḥ*

what.SG.N 3SG.M.DST.INS EXPL production.INS mother.SG.GEN

yauvana-hāri-ṇā?   ārohati   na yaḥ   svasya

youth-depriver-SG.M.INS climb.3SG.PRS not REL.3SG.NOM self.SG.M.GEN

vaṃśa-sya   agre   dʰvajaḥ yatʰā.

lineage-SG.M.GEN before flag   as

'What is the use of the birth of a son, the destroyer of his mother's youth, who does not rise up to his family like a flag?' <kīlotpāṭi-vānarakatʰā, mtbd>

In the example above *yaḥ* and *svasya* are co-referential and *svasya* is modifying the noun *vaṃśasya* as is clear from their agreement. In the following instance *svam*, the accusative form of *sva*, modifies *durgam* and co-refers with *caurasiṃhaḥ*:

(58) *tvām   dr̥ṣ-ṭvā   dūrataḥ   api   **caura-siṃha-ḥ**   praviṣ-ṭaḥ   <u>*svam*</u>*

2sg.acc see-GRN distantly EMPH thief-lion-SG.NOM enter-PSPL self.M.SG.ACC

<u>*durgam.*</u>

fort.SG.ACC

'Seeing you distantly, the thief lion entered in his fort.' <siṃha-śaśaka-katʰā, mtbd>

(59) *śrūy-ate   hi   **kapot-ena**   śatruḥ   śaraṇam   āga-taḥ*

hear-3SG.PRS.PASS EXPL pigeon-SG.INS enemy shelter come-PSPL

*pūji-taḥ   ca   yatʰā-nyāyam   <u>**sv-aiḥ**</u>   ca   <u>māṃs-aiḥ</u>   nimantri-taḥ*

worship-PSPL and as-justice   self-M.PL.INS and flesh-M.PL.INS invite-PSPL

'It is heard that the enemy in the shelter of a pigeon was worshiped properly and served by his own (pigeon's) flesh.' <haimahaṃsa-katʰā, kkly>

(60) ***śaśak-āḥ**   ca   tat   din-āt   ārabʰ-ya   sa-parivārāḥ   sukʰ-ena*

rabit-PL.NOM and that day-SG.ABL start-GRN with-family pleasure-SG.INS

<u>*sv-eṣu*</u>   *stʰāneṣu   tiṣṭʰ-anti   sma.*

self-N.PL.LOC place.N.PL.LOC sit-3PL.PRS PST

'Since that day, the rabbits were started living happily in their places.' <śaśaka-gajayūtʰapa-katʰā, kkly>

sva is synonymous with the word relative. In the following sentence it has been used for close relatives:

(61) *daurgatyam dehin-ām   duḥkʰam apamānakaram varam*

poverty   people-PL.GEN pain   insult-doer   great

*yena   **sv-aiḥ**   api   man-yante   jīvantaḥ   api   mr̥tāḥ iva*

COMP self-M.PL.INS EMPH assume-3PL.PRS live-PRPL.PL EMPH dead like

'Poverty of people is greatly painful and insulting. Even own relatives consider them as dead when they are alive.' <śabaraśūkara-katʰā, mtsp>

2.2.1.3.2. svakam

svakam is a variant of *svam*, but the *-ka* element does not serve any purpose here. *sva* and *svakam* are semantically same. In Sanskrit morphology *-ka* is a suffix which is sometimes attached to some lexemes, but remains semantically vacuous, as the meaning of the resultant word is the same. In the following sentence *svakam* has been used as the modifier of *putram*, and its antecedent is *yoṣitām*:

(62) āstām-tāvat kim any-ena   daurātmy-ena   iha   yoṣit-ām

     let_it_be what other-SG.INS wickedness.SG.INS here women-PL.GEN

     vidʰṛ-tam sva-udareṇa   api gʰnanti   *putram*   *svakam*

     bear-F.PSPL self-stomach.INS EMPH kill.3PL.PRS son.M.SG.ACC self.M.SG.ACC

     ruṣā.

     anger.INS

     'Let this be aside, why talk of any other wickedness of women? They kill, through anger, even the child of their womb.' <kāmāturadʰanavāna-katʰā, ldpr>

2.2.1.3.3. svakīya

(63) tatʰā anuṣtʰ-ite   *devaśarmā*   api vitta-nāśa-samudbʰūta-śoka-rahita-ḥ

     same happen-PSPL.LOC Devasharma.NOM too wealth-loss-caused-agony-less-NOM

     punaḥ api *svakīya-m* *matʰāyatana-m* jagāma.

     again emph self-N.SG.ACC monastery-N.SG.ACC reach.PST

     'After this happenning, Devasharma having forgotten his agony out of the loss of wealth, returned to his own place.' <dūtijambūkāṣādʰabʰūti-katʰā, mtbd>

(64) tat tvam ulūka-yoni-ga-taḥ   api *svakīyām*

     then you owl-species-go-PSPL EMPH self.F.SG.ACC

     eva *vāyasa-yonim*   bahu manya-se.

     only crow-species.F.SG.ACC very consider-2SG.PRS

'Then, even after going in the owl's species, you consider your own species of crow greater.' <ratʰakāravadʰū-katʰā, kkly>

In the text there is no example for the independent use of *svīya* to quote here. This has been only used as a compound reflexive, so that use would be given in the next section.

2.2.1.3.4. *ātmīya*

(65) *param aham   nad-yām   snāna-artʰam gam-iṣyāmi tat   **tvam***
       CONJ 1SG.NOM river-SG.F.LOC bath-for   go-1SG.FUT then 2SG.NOM

       ***ātmīyam*   śiśum   enam   dʰanadevanāmāna-m   mayā**
       self.M.SG.ACC child.M.SG.ACC 3SG.ACC.PRX Dhanadeva-named-SG.ACC 1SG.INS

       saha snāna-upakaraṇa-hasta-m   preṣaya   iti.
       with   bath-apparatus-hand-M.SG.ACC send.2SG.IMP QUOT

       'I will go to take bath in river, so you send your child namely Dhanadeva along with bathing stuff with me.' <lohatulā-vaṇikputrakatʰa, mtbd>

This *ātmīya* possessive reflexive may have negative form as well. There is one such use in the text. In such usage the common negative prefix un- is prefixed.

(66) *muhūrtam api   **an-ātmīyam**   bhoktum na   labh-yate*
       moment   EMPH not-self-N.SG.NOM eat.INF not attain-3SG.PRS.PASS

       'You can't have it even for a moment if it is not yours.' <vaṇikputra-katʰā, mtsp>

2.2.1.3.5. *nija*

(67) *yat-artʰam **bʰrātar-aḥ**   **putr-āḥ**   api vāñcʰ-anti ye*
       which-for brother-M.PL.NOM son-M.PL.NOM EMPH wish-3PL.PRS REL.3PL

       ***nijāḥ**   vadʰam rājya-kr̥-tām   rājñ-ām   tat rājyam*
       self.M.PL.NOM killing state-do-PSPL.PL.GEN king-PL.GEN that state.N.SG.ACC

       dūrataḥ tyajet.

45

distantly renounce

'The kingdom for which own brothers and sons want to kill the king, one should renounce that kingdom distantly.' <mantʰarakaulika-katʰā, apkr>

The sentence below demonstrates the agreement of neuter, plural adjective and *nija*.

(68) *ye   ca   prāhuḥ   duḥ-ātmānaḥ duḥ-ārādʰy-āḥ*
     REL.PL.NOM and say.PL.PRS bad-souls   difficult-worshipable-NOM.PL
     mahībʰujaḥ   pramāda-ālasya-jāḍy-āni   kʰyāpit-āni
     kings   carelessness-laziness-foolishness-N.PL report.PSPL-N.PL
     nijāni   taiḥ
     self.N.PL 3SG.PL.INS

'The people who say kings are wicked, and difficult to be pleased; carelessness, laziness, and foolishness are advertised by them only.' <kīlotpāṭi-vānarakatʰā, mtbd>

(69) *tat   adya enam bhaya-ākulam   prāpya   sva-budʰi-prabʰāv-eṇ*
     then today this fear-perturbed received self-wisdom-impact-INS
     niḥ-bʰayam   kṛ-tvā vaśīkṛtya   ca   nij-ām   sācivya-padavī-m
     without-fear do-GRN controlled and self-F.SG.ACC secretary-post.F-SG.ACC
     sama-āsāday-iṣyāmi.
     well-attain-1SG.FUT

'Then, today getting this scared, making him fearless by the influence of my intelligence, and making him enslave, I will attain my secretary post.' <kīlotpāṭi-vānarakatʰā, mtbd>

2.2.1.3.6. *ātman* as a Possessive Reflexive

As has been said earlier, the genitive inflections of *ātman* behave like possessive reflexives, but they do not agree with the possessed item. Here are some examples:

(70) *yogya-ḥ   ayam   dr̥ś-yate   tat   kat^ha-yāmi etasya*

worthy-SG.NOM 3SG.M.PRX.NOM look-3SG.PRS then say-1SG.PRS 3SG.M.PRX.GEN

agre   ātman-aḥ   ab^hiprāya-m.

before self-SG.GEN intention-SG.ACC

'This seems to be a worthy person so I will say my intention to him.' < kīlotpāṭi-

vānarakat^hā, mtbd>

(71) *a-vidi-tvā   ātman-aḥ   śaktim   para-sya   ca   samutsuka-ḥ.*

not-know.GRN self-SG.GEN   power.F.SG.ACC other-SG.GEN and curious-SG.NOM

gacc^h-an   ab^himuk^haḥ nāśa-m   yā-ti   vahn-au   pataṅga-vat.

go-PRPL.SG towards   destruction-ACC go-3SG.PRS fire-SG.LOC insect-like

'Without knowing enemy's and self's power, if a person curiously goes to battie

field, faces destruction like an insect in fire.' <siṃha-śaśaka-kat^hā, mtbd>

(72) *tat   anviṣ-yatām yat^hā asmāt   vyasan-āt   ātman-ām su-nirmuktiḥ.*

then search-IMP so_that 3SG.PRX.ABL addiction-ABL self-PL.GEN well-salvation

'Then search first, so that we can get rid of this addiction.' <śaśaka-gajayūt^hapa-

kat^hā, kkly>

2.2.1.4. *Compound Reflexives*

All the possessive reflexives, like *ātma, nija, sva, svakīya, svīya, (ātmīya* in compound is
not available in the text) and the intensive reflexive *svayam* are compounded with nouns
or participles. These reflexives in their root form are compounded with the possessed
noun/participle and constitute the first component of such compounds.

2.2.1.4.1. *ātma*

Following is an example for the compounding of reflexive *ātman* with the noun *vedanā*.
In the construction *ātman* is without any casal inflection and is placed in the first
component of it. The noun *vedanā* is in accusative case.

47

(73) *vayasya yadi evam tat śru-ṇu   me   rahasyam   yena*

friend, if   thus then listen-2SG.IMP 1SG.GEN secret.ACC REL.3SG.INS

*sarv-ām   **ātma-vedan-ām**   te   vad-āmi.*

all-SG.F.ACC self-agony-SG.F.ACC you.DAT tell-1SG.PRS

'Friend, if this is so, then, you listen to my secret by which I tell you all the agony of mine.' <viṣṇurūpadʰṛkkaulika-katʰā, mtbd>

In the sentence below *ātman* is compounded with a past participle.

(74) *bʰoḥ mitra! vārdʰakabʰāv-e kim tvam   **ātma-viceṣṭ-itam** smar-iṣyasi?*

Oh friend old_age-LOC what 2SG.NOM self-do-PSPL.ACC remember-2SG.FUT

'Oh friend! during old age what work would you remember done by you?' <dʰarmabuddʰi-pāpabuddʰi-katʰā, mtbd>

In the language *ātman* is compounded with the morpheme *-ja* 'born' also, for making the word *ātmaja*, meaning a son. The word *ātmaja* inflect for feminine gender and has the form *ātmajā* to denote daughter.

(75) *na mātar-i   na dār-eṣu   na sodary-e   na ca   **ātma-j-e***

not mother-LOC not wife-PL.LOC not brother-LOC not and self-born-M.SG.LOC

viśrambʰaḥ tādṛśaḥ   puṃs-ām yādṛk mitr-e   nirantar-e

confidence like_that men-GEN as   friend-LOC constant-LOC

'The men trust their close friend to such an extent, that they do not trust their mother, wife, brother, and son in the same way.' < vṛṣabʰānugaśṛgāla-katʰā, mtsp>

2.2.1.4.2. *nija*

The pattern of compounding in the following example is: *nija*+noun.

(76) tat   sambod^hyaḥ adya tvayā   **nija-b^hartā** yat^hā mama śatrūn

then address.GRN today 2SG.INS self-husband as   1SG.GEN enemy.PL.ACC

vyāpādayati

kill.3SG.PRS

'Then, you should address your husband today, so that he would kill my enemies.'

<viṣṇurūpad^hṛkkaulika-kat^hā, mtbd>

In the following instance the pattern is: *nija*+participle.

(77) ....*yena*   **nija-vihitam** *iyam eva* *b^huṅkte*

so that self-do.PSPL   this.F EMPH   experience.3SG.PRS

'........ So that, she experiences her deeds only.' <valmīkodarast^hasarpa-kat^hā,

kkly>

2.2.1.4.3. *sva*

In the sentence below the pattern of compounding is: *sva*+noun.

(78) *tataḥ vivāhānantaram rājā sāntaḥpuraḥ* **sva-gṛham** *ānīya*

then marriage-after king with-queen self-home arrive.PSPL

ab^hyarcitaḥ

worship.PSPL.SG

'Then after marriage, the king arrived his home with his queens, and was

worshiped.' <dantilagoramb^hayoḥ kat^hā, mtbd>

sva can be used in reduplication also. In the following sentence *sva* has been reduplicated
and then finally compounded with *jana* (people).

(79) *sāmpratam sarveṣām*   *svalpa-jalāśayānām jalacarāḥ*   *gurujalāśayeṣu*

now   all.M.PL.GEN small-pond.PL.GEN aquatic_creature.PL big-pond.PL.LOC

sva-sva-janaiḥ   *nīyante.*

self-self-people.PL.INS escort.PL.PRS.PASS

'Now, the aquatic creatures of all the small ponds are being escorted by their relatives to the big ponds.' <bakakarkaṭaka-katʰā, mtbd>

2.2.1.4.4. svīya

svīya in the text is compounded with noun only, following is an example:

(80) atʰa prāptavyam artʰena api **svīya-pitṛmātarau** samasta-kuṭumba-

after Praptavyam Artha.sg.INS also self-father-mother.DU all-family-

āvṛtau tasmin nagare saṃmānapuraḥsaram sam-ānītau

surrounded.DU 3SG.DST.LOC city.SG.LOC respectfully well-bring.PSPL.DU

'Aftewards, his family including his parents and all were brought respectfully to that city by Praptavyam Artha.' <vaṇikputra-katʰā, mtsp>

2.2.1.4.5. svakīya

Like svīya, svakīya is also compounded with noun only in the text. Following is an example:

(81) yadi brāhmaṇa! tvam **svakīya-jīvita-sya** arddʰa-m dad-āsi tataḥ

if Brahmana.VOC! 2SG.NOM self-life-GEN half-ACC give-2SG.PRS then

te jīva-ti brāhmaṇī.

2SG.GEN live-3SG.PRS Brahmani

'Hey Brahaman, if you give half of your life, then your Brahmani would be alive.'
<brāhmaṇadampatyoḥ katʰā, ldpr>

2.2.1.4.6. svayam

The intensive reflexive svayam can be compounded as well. Following verse is such an example where it has been compounded with a paparticiple.

(82) jambuka-ḥ huḍu-yuddʰ-en vayam ca āṣādʰabʰūti-nā.

jackal-SG.NOM sheep-battle-SG.INS 1PL.NOM and Ashadhabhuti-SG.INS

50

dūtikā   para-kāry-eṇa   trayaḥ   doṣ-āḥ   svayaṃ-kṛ-tāḥ.

Dutika.NOM   other-work-SG.INS   trinity   fault-PL.M   self-do-PSPL.PL

'The jackal was killed in the battle of sheep, we were destroyed by Ashadhabhuti,

and Dutika was killed by weavers. These three made their own destruction.'

<dantila-gorambʰayoḥ katʰā, mtbd>

2.2.2. Reciprocals

2.2.2.1. *Bare Reciprocals*

Reciprocal pronouns do not inflect for case, gender, number, or person; however, there is

one instance of *parasparam* that has been used in genitive singular. All the reciprocals

always need either a plural antecedent or a dual one for their interpretation. The

reciprocal pronouns used in the text PT include *anyōnyam* (occured once), *parasparam*

(most frequent), *itaretaram* (occured once), and *mitʰaḥ* (most frequent). Following are the

examples for reciprocal usage taken from the text:

(83)   *tau   ca   parasparam   mantra-yataḥ*

    3DU.M.DST.NOM   and   REC   advise-3DU.PRS

    'And they (two) advise each other.'   <prastāvanā-katʰā, mtbd>

In the example above the reciprocal *parasparam* has a dual antecedent and in the

following senetece the antecedent is in plural:

(84)   *atʰa   anyasmin   divase   te   brāhmaṇāḥ   parasparam*

    afterwards   other.SG.LOC   day.SG.LOC   3PL.DEF.NOM   brahmana.PL.NOM   REC

    niścayam   kṛ-tvā   vidyā-upārjana-artʰam   kānyakubja.e   gatāḥ.

    confirm   do.GRN   knowledge-gain-for   Kanyakubj-SG.LOC   go-PSPL.M.PL

    'Afetrwards, on the other day, those brahamanas after mutually decided, went to

    Kanyakubja for gaining knowledge.'   <mūrkʰapaṇḍita-katʰā, apkr>

In the following sentence the genitive form of *paraspara* has been used. However, this is single instance of this kind in the text, where it has inflected for a case. Its inflection for other cases is completely absent.

(85) *evam* **paraspara-sya** *marm-āṇi*   *udgʰāṭita-van-tau.*
      thus REC-SG.GEN   secret-N.PL.ACC reveal-PSPL-DU
      'Thus, they revealed the secrets of each other.' <valmīkodarastʰasarpa-katʰā, kkly>

But following is a counter example of the above where *parasparam* could have been used in genitive but it remains stagnant and gives the meaning of genitive.

(86) *evam te*   *trayaḥ api* *śiśavaḥ* **parasparam** *a-jñāta-*
      thus 3PL.DEF.NOM three EMPH children REC   not-known-

      jāti-viśeṣ-āḥ   *eka-āhāra-vihār-āḥ*   *bālya-samayam*
      caste-particular-M.PL same-food-pastime-M.PL childhood-time

      nirvāha-yanti   sma.
      spend-3PL.PRS PST
      'Thus, those three children unknowing each other's castes, were spending their childhood having same food and pastime.' <siṃha-śṛgālaputrayoḥ katʰā, ldpr>

In the sentence below the antecedent *tābʰyām* is in dual dative form but the reciprocal *anyonyam* remains intact.

(87) *mayā* **anyonyam** *tābʰyām*   *mitʰyā-prajalpan-ena bʰedaḥ*   *tatʰā vihitaḥ,*
      1SG.INS REC   3DU.M.DST.DAT false-gossip-M.SG.INS difference so   done

      yatʰā bʰūyaḥ api mantra-yantau   eka-stʰāna-stʰi-tau   *na drakṣya-si.*
      as   again also advise-PRPL.M.DU one-place-place-PSPL.M.DU not see-2SG.FUT
      'The big difference has been created between them by me through false gossip, so, you will not see them (two) again in one place advising each other.' <caṭaka-kuñjara-katʰā, mtbd>

52

In the example below the reciprocal *mitʰaḥ* refers to the antecedent which is in dual form.

(88) *tataḥ   ca   niśītʰ-e   yāvat paśya-ti tāvat tau   eva*

afterwards and midnight.LOC REL   see-3SG.PRS then 3DU.DST.M.NOM only

dvau puruṣ-au   mitʰaḥ mantra-yataḥ.

two man-DU.NOM REC   advise-3DU.PRS

'And afterwards, when in the midnight he sees, those two men advise each other.'

<vṛṣabʰānugaśṛgāla-katʰā, mtsp>

The sentence (89) serves as the example for the antecedent in plural form and passive construction for *mitʰaḥ*.

(89) *taiḥ   ca   tādṛśam pīvara-m paśu-m   skandʰ-e*

3PL.DST.M.INS and like_that fat-SG.ACC animal.SG.ACC shoulder-SG.LOC

ārūḍʰa-m   avalok-ya   mitʰaḥ abʰihitam

climbe.pspl-acc behold-GRN REC   say.PSPL

'On seeing that fat animal climbed on the shoulder it was said by them to each other.' <dʰūrtabrāhmaṇacʰāga-katʰā, kkly>

The following verse is a beautiful example of reciprocal use where two reciprocals *parasparam* and *itaretaram* have been used and they demonstrate that both are semantically the same.

(90) *yatʰā cʰāyā-tap-au   nityam su-sambaddʰ-au   parasparam*

as   shadow-sun-DU always well-affiliated-DU.M REC

evam   karma ca kartā   ca saṃ-śliṣṭ-au   itaretaram

same_way act.NOM and doer.NOM and well-stick.PSPL-DU REC

'As shadow and sun are always well connected to each other, so are the act and doer to each other.' <mandabʰāgyasomilaka-katʰā, mtsp>

53

2.2.2.2. *Compound Reciprocals*

Like reflexives, reciprocals also undergo compounding. However, in the text only *parasparam* was found in compounding constructions. In compounding it loses the coda of its final syllable, that is, *m*.

(91)   *yataḥ*   *rakṣit-en*   *anena*   *kadācit*   **paraspara-prīt-yā**

        because   protect.PSPL-SG.INS   3SG.PRX.INS   perhaps   mutual-love-SG.F.INS

        kālaḥ   *sukh-ena*   *gacchᵃ-ti.*

        time.M.SG.NOM   pleasure-SG.INS   go-3SG.PRS

        'Because, from its protection, perhaps, it is possible to spend time pleasantly with mutual love. (literal)' <brāhmaṇacaurapiśāca-kathᵃā, kkly>

(92)   *na*   *vinā*   *pārthiva-ḥ*   *bʰṛty-aiḥ*   *na*   *bʰṛty-āḥ*   *pārtʰiva-m*

        not   without   king-SG.M.NOM   servant-PL.INS   not   servant-PL.NOM   king-SG.ACC

        vinā   *teṣām*   *ca*   *vyavahāraḥ*   *ayam*   **paraspara-nibandʰanam**

        without   3PL.M.DST.GEN   and   behaviour   3SG.M.PRX   mutual-binding

        'There is no king without servants and there are no servants without a king. Their relation binds them with each other.' <kīlotpāṭi-vānarakathᵃā, mtbd>

2.3. Cataphoric Expressions

Cataphoric expressions are those where the pronoun precedes the linguistic expression necessary for its interpretation in the given discourse. In PT these expressions have been sparsely used. Some of such usages are illustrated through the following sentences:

(93)   *tat*   *śru-tva*   *ga-ta-āyuṣa-m*   *iva*   *ātmān-am*   *manyamānaḥ*   **saṇjīvaka-ḥ**

        that   listen-GRN   go-PSPL-age-ACC   as   self-ACC   assuming   Sanjeevaka-NOM

        param   *viṣāda-m*   *agamat.*

        great   sadness-ACC   go.3SG.PST

        'After listening that, Sanjeevaka believing himself as dead, got to great sadness.' <śṛgāladundubʰi-kathᵃā, mtbd>

(94) *sevaka-ḥ   svāmin-am   dveṣ-ṭi   kṛpaṇa-m   paruṣākṣāra-m*

servant-M.NOM   master-SG.ACC   hate-3SG.PRS   miser-SG.ACC   harsh-ACC

ātmān-am *kim* ***saḥ***   *na   dveṣ-ṭi   sevya-a-sevya-m   na*

self-ACC   why   3SG.M.NOM   not   hate-3SG.PRS   doable-not-doable-ACC   not

vet-ti   ya-ḥ

know-3SG.PRS   REL-3SG.NOM

'If a servant hates his miser and harsh master, then why he does not hate himself,
who does not know who is servable and unservable.' <kīlotpāṭi-vānarakatʰā,
mtbd>

In the following example the reciprocal *parasparam* is occurring before its antecedent
karaṭakadamanakau, and thus presents an instance of cataphoric usage of reciprocal:

(95) *tataḥ svami-prasāda-rahit-au*   *kṣut-kṣāma-kaṇtʰ-au*   ***parasparam***

then   master-mercy-less-DU.NOM   hunger-weak-throat-DU.NOM   REC

karaṭaka-damanak-au   *mantra-yete.*

Karataka-Damanaka-DU.NOM   advise-3DU.PRS

'Then, deprived of the mercy of their master, Karataka and Damanaka advise each
other.' <dantilagorambʰayoḥ katʰā, mtbd>

Thus, in the text both reflexives and reciprocals have occurred as capahors too, though
the frequency of such uses is considerably low, as is common in other human languages
as well.

Chapter 3

SARS Design: Issues and Challenges

SARS is a design and development of a computer program for identifying the antecedents of reflexive and reciprocal pronouns in a Sanskrit sentence. It is implemented in Java and JSP and runs on PCs on Windows environment (see Chapter 4). This chapter discusses the problems encountered in developing the program for automatic resolution of anaphors. Human languages are intrinsically complicated. and Sanskrit is no exception to it. The anaphoric aspect of Sanskrit is full of complexities as we have seen in the Chapter 2. Even the humans have problems in pairing the anaphors and their antecedents while reading the Sanskrit texts due to its relatively free word ordering and various other stylistic complexities. After discussing these issues, which are important to consider before designing a robust system for this task. we propose an algorithm for handling reflexives and reciprocals automatically.

3.1. Complexity of Sanskrit Texts

In this section we discuss the characteristics of the Sanskrit language and its texts which are relevant for our purpose. There is a range of complexities in the language. but we selectively discuss those only which affect the design of the system.

3.1.1. Irregularity of Punctuation Marks

The punctuation marking in Sanskrit texts is bizarre; they do not use any kind of reliable punctuation. Originally. Sanskrit had no punctuation. In the 17th century. Sanskrit and Marathi. both written in the Devanagari script. started using the vertical bar "ǀ" (single *daṇḍa*. also called *pūrṇavirām* in Hindi)[2] to end a line of prose and double vertical bars "ǁ" (double *daṇḍa)* in verse (Wikipedia). In unpunctuated texts. the grammatical structure of sentences in classical writing is inferred from the context. Sanskrit by itself contains

---

[2] There is a common practice in linguistic literature of Sanskrit written in English to refer the "ǀ" sign as *daṇḍa* (see Huet 2009 and Hellwig 2007). In common parlance. however. it is called *pūrṇavirām* also.

only "।" to indicate an end of a sentence or half of a verse and "॥" to indicate the end of a complete verse. However, with the advent of the printed books, most punctuation marks used in English are also being used in printed Sanskrit texts. Punctuation sometimes plays a significant role in understanding the text and misplacement of a punctuation mark can reverse the intended meaning. Thus punctuations are important features of a text and they certainly help in following the text.[3]

The implementation of punctuation marks in Sanskrit texts has been rather irregular and complex. These kinds of irregularities are visible in PT too. A sample is being given here from "mitrasaṃprāptikam" (the second section of PT) from Shrishyamacharan Pandey's recension (2006:237):

तत्र च लघुपतनको नाम वायसः प्रतिवसति स्म । सः कदाचित्प्राणयात्रार्थं पुरमुद्दिश्य प्रचलितो यावत्पश्यति, तावत् जालहस्तोऽतिकृष्णतनुः, स्फुटितचरणः, ऊर्ध्वकेशो यमकिङ्करराकारो नरः संमुखो बभूव ।

'A crow called Laghupatanak was living there. One day when he was going to the city in quest of food, he saw a man passing before him who was with a net in his hands, dark colour, splay-footed, hair raised up, and looking like the servant of Yama (the god of death).'

---

[3] The lack of punctual punctuation in our text culture was rightly observed by the renowned novelist Raja Rao in his foreword to his novel *Kanthapura* (1938): "After language the next problem is that of style. The tempo of Indian life must be infused into our English expression, even as the tempo of American or Irish life has gone into the making of theirs. We, in India, think quickly, we talk quickly, and when we move we move quickly. There must be something in the sun of India that makes us rush and tumble and run on. And our paths are paths interminable. The *Mahabharatha* has 214778 verses and the *Ramayana* 48000. Puranas there are endless and innumerable. We have neither punctuation nor the treacherous 'ats' and 'ons' to bother us – we tell one interminable tale. Episode follows episode, and when our thoughts stop our breath stops, and we move on to another thought. This was and still is the ordinary style of our story-telling. I have tried to follow it myself in this story...."

In sandhi places it is not necessary that the two words would be concatenated in writing too; they might be written separately too, depending on the nature of the sandhi. In the given sample due to sandhi, commas have not been put in the two requiring places: जालहस्तोऽतिकृष्णतनुः and ऊर्ध्वकेशो यमकिङ्कराकारो; because, then, it would have invited the sandhi-splitting which the editors of texts do not do. But the computational linguists have to split them up in order to process the language. The sandhi free version of the above text would look like below. Herein, the two above mentioned places have been given commas rendering the text punctuationally punctual. This was possible only when the optional or external sandhis were split and the internal sandhis kept intact. The concerned places are underlined.

तत्र च लघुपतनकः नाम वायसः प्रतिवसति स्म । मः कदाचित् प्राणयात्रार्थम् पुरम् उद्दिश्य प्रचलितः यावत् पश्यति, तावत् जालहस्तः, अतिकृष्णतनुः, स्फुटितचरणः, ऊर्ध्वकेशः, यमकिङ्कराकारः नरः संमुखः बभूव ।

This punctuational disorder creates problems in identifying sentence boundaries and clause boundaries which are very crucial for anaphora resolution system design. Hellwig (2007:38) observes:

> *dandas* may be helpful in generating hypotheses about the syntactic structure of a text, but cannot be considered as punctuation marks in a strict sense. This lack has a far reaching effect on any tagging or parsing process applied to a Sanskrit text, because it cannot be guaranteed that all words necessary for a complete analysis are really contained in the text delimited by these marks.

The text PT is full of verses. The completion of verse, as stated earlier, is marked by double *dandas* and this marking has been a great problem in tokenizing the text. The double *dandas* are used in headings also and are typical style of Sanskrit text writing. In tokenization the double *dandas* were first replaced by single *dandas* and then the text was tokenized basing the delimitation on the single *danda*.

3.1.2. *Sandhi* Phenomenon and the Writing Convention

Sandhi[4] phenomenon is a prime feature of Sanskrit language. It is a highly synthetic language and the word boundaries in spoken as well as in written forms are often faded away due to intense concatenation. To identify word boundaries, especially in written form, is not an easy task. There are, actually, two kinds of phenomena involved in these kinds of euphonic combinations. They are commonly described as external or *anitya* sandhi and internal or *nitya* sandhi. When we split the external sandhis the components remain usable in the sentence; they do not require inflectional suffixes as they are already endowed with before and after sandhi-ing, whereas this is not the case with internal sandhi. To identify *nitya* and *anitya* sandhi in Sanskrit there is a famous and well established rule composed in the following *kārikā* (doctrine stated in verse):

karikā (1) *saṃhitaikapade nityā nityā dhātūpasargayoḥ* I

nityā samāse vākye tu sā vivakṣāmapekṣate II

'In a *pada* (roughly a word) a *sandhi* is mandatory and also in combination of prefixes and *dhātus* (verb roots). It is mandatory in compound constructions also, but in a sentence it requires the intention of the speaker.'

Thus, barring the mandatory or *nitya* sandhi cases, the rest instances of sandhi are completely dependent on the speaker or writer whether they wish to combine two or more *padas* or not. And this kind of sandhi, viz. external sandhi clearly involves combination of two or more *padas*. The idea in the above *kārikā* beautifully works in identifying internal and external sandhis in sandhi-splitting tasks.

Sandhis, especially the external ones, are serious obstacles to an easy tokenization of Sanskrit texts. The unpreprocessed text has a lot of problems in identifying word

---

[4] Sandhi means euphony transformation of words when they are consecutively pronounced. Typically when a word w_1 is followed by a word w_2, some terminal segment of w_1 merges with some initial segment of w_2 to be replaced by a "smoothed" phonetic interpolation, corresponding to minimising the energy necessary to reconfigure the vocal organs at the juncture between the words (Huet, 2006).

59

boundaries. The external *sandhis* have to be resolved first for smooth POS tagging and for anaphora resolution also. To unglue each *padas* from euphonic combinations is itself a complex process which requires the identification of external (*anitya*) sandhis and internal ones in the text and the sandhi-splitting rules available in the grammar. For sandhi-splitting a system has been designed at HCU, Hyderabad. For POS tagging these cases of *anitya* sandhi must be resolved first. To tag a sequence of words with *anitya* sandhis is impossible. After splitting *anitya sandhis* the *kārikā* (1) will look like the following:

saṃhitā ekapade nityā nityā dhatūpasargayoḥ ।

nityā samāse vākye tu sā vivakṣām apekṣate ॥

In this condition of the verse each *pada* is standing alone, and now can be tagged easily. In our tagging scheme each *pada* is tagged separately (Jha et al. 2009, and Gopal et al. 2010).

The orthographic system of Sanskrit language is rather complex and this is not only obstacle for anaphora resolution but also for many other NLP tasks. Due to this complexity of the language sometimes two or more *padas* are concatenated and they seem to be one word, but actually they are not. And to separate a *pada* (a usable unit in a sentence) from other adjacent *padas* is sometimes not an easy task. Some systems have been developed for such tasks. This generally happens with consonant ending words followed by words having vowel in their initial position. These cases as per orthographic rule - *ajjhīnam pareṇa saṃyojyam* 'the vowel-less segment should be attached to the following vowel' – are concatenated, that is, the ending consonant of preceding word hosts the starting vowel of the following word. For POS tagging of the text this concatenation has to be broken up.

3.1.3. Pronoun Compounding

As is discussed in the second chapter, Sanskrit pronouns are sometimes compounded with nouns and participles. We have seen that roots of pronouns, possessive pronouns of all persons, and roots of lexical anaphors (reflexives and reciprocals) undergo compounding. In the process of compounding only root of the pronoun is left and the case, number, and gender features are dropped. And, this creates problems in identifying their antecedents, as the grammatical features are main clues for pairing anaphor and antecedents in our approach. These kinds of cases demand different treatment, like the compound processor and then POS tagger which are complex things to be done. For example in (96) *sva-gṛham* is such a compound; it is as a whole pada and in the present situation this would be tagged as common noun with grammatical features. Now, without splitting this compound there is no way to recognize the reflexive *sva*. To enable the system to recognize it, one has to get this compound split and then transform the components into *padas* and then get them POS tagged separately. This extra work has to be done manually or automatically to get the *sva* out from the compound. After the POS tagging the system would be able to find its referent.

(1) *tataḥ vivāhānantaram rājā sāntaḥpuraḥ sva-gṛham ānīya*
     then marriage-after king with-queen self-home arrive.PSPL
     abʰyarcitaḥ
     worship.PSPL.SG
     'Then after marriage, the king arrived home with his queens, and was worshiped.'
     <dantilagorambʰayoḥ katʰā, mtbd>

Apart from this, the compounding of other categories also is a hurdle in the processing of the language. Its compounding nature slows down any computational analysis of the language. To carry out the processing of the language smoothly one has to have efficient systems for compound analysis and sandhi analysis.

3.1.4. Multiplicity of Pronoun Forms

A slightly uneasy problem in Sanskrit anaphora resolution is that Sanskrit has pronoun forms in abundance. Apart from the regular inflections of pronouns, the language has many strategies to encode pronominal information, and this has been seen in the second chapter of this dissertation. There are 105 personal pronoun forms (four of them are homophonous with linkers), 19 reflexive forms, and 5 reciprocal forms available in PT itself, and they are more in the language. One has to consider this large number of pronoun forms found in the language while designing a computer program. Moreover, many forms are instances of syncretism and they need to be dealt with carefully in order to have precise results. This richness increases the load of the system and the researcher as well. One has to take care of all the regular forms and all the possible forms generated by the extraordinary strategies of the language. For instance, the following sentence (2) a. can be equivalently expressed by any of the forms given below. All of them are semantically identical.

(2) a. *rāma-ḥ   sva-m   pustakam   paṭʰati.*

   rāma.NOM   self-ACC   book-ACC   read-3SG.PRS

   'Ram reads/is reading his book.' <self made, as per my intution>

b.   *rāmaḥ svasya pustakam paṭʰati.*

c.   *rāmaḥ sva-pustakam paṭʰati.*

d.   *rāmaḥ svakīyam pustakam paṭʰati.*

e.   *rāmaḥ svakīya-pustakam paṭʰati.*

f.   *rāmaḥ svakam pustakam paṭʰati.*

g.   *rāmaḥ svīyam pustakam paṭʰati.*

h.   *rāmaḥ svīya-pustakam paṭʰati.*

i.   *rāmaḥ ātmanaḥ pustakam paṭʰati.*

j.   *rāmaḥ ātma-pustakam paṭʰati.*

k.   *rāmaḥ ātmīyam pustakam paṭʰati.*

l.   *rāmaḥ ātmīya-pustakam paṭʰati.*

m.   *rāmaḥ nijam pustakam paṭʰati.*

n.   *rāmaḥ nija-pustakam paṭʰati.*

o.   *rāmaḥ nijasya pustakam paṭʰati.*

These are the multiple ways of expressing possessive reflexive in Sanskrit. Likewise, other kinds of pronouns also have this kind of richness some or other way. Their morphological and syntactic analysis has to be done thoroughly in order to train the machine efficiently.

3.1.5. Word Order in Sanskrit

Reference relations may be constrained by the syntactic relationships between a referential expression and a possible antecedent when both occur in the same sentence (Jurafsky and Martin, 2000). Syntactic relationship is determined by the position of the words in a sentence in configurational languages whereas in non-configurational languages the position of words does not matter. Languages with rigid word order are called configurational languages and languages which have relatively free word order are termed as non-configurational languages (Khanna, 1996). Sanskrit and many other Indian languages show very strong agreement between verb and the subject of the verb. The person and number features of the subject are encoded in the verb also and that is why sometimes the subject is dropped, as the verb gives information about the subject also (see 3.1.7. also). It is generally agreed that languages which exhibit the phenomenon of case assigning and case marking exhibit free word order and languages which exhibit only case assigning phenomenon without case marking do not display free word order. Sanskrit has relatively free word order, so a non-configurational language. Virtually, any *pada* (a usable linguistic unit in a sentence) can be put any where, however, the ordering of constituents in a compound can trigger change in meaning. For instance, *rāja-puruṣa* ("king's man") does not mean the same as *puruṣa-rāja* ("man's king"). Indeed, only in very special circumstances can two words in a Sanskrit compound be permuted without disturbance of meaning (Gillon, 1996).

Pāṇini gives many rules for the order of parts of words (e.g. stem, suffix) and for the order of members of compounds, which are also parts of words since nominal compounds

63

are treated as nouns. He nowhere seems to give any rule for the order of words in the sentence. But this does not imply that he gives no rules for the relation of words in the sentence (Ananthanarayana, 1996). His *kāraka* theory is, actually, for describing the relations of words in a sentence.

On the issue of word order in the language the modern Sanskrit scholar and speaker of the language Apte in his *The Student's Guide to Sanskrit Composition* (1934) states:

> "In Sanskrit every word (except adverbs and particles) is inflected and the grammatical inflection itself shows the relation in which one word stands to another. Thus, GRAMMATICALLY speaking, there is no order as such that need be much attended to...But if there is no GRAMMATICAL order, there is a sort of logical sequence of ideas, which must follow one another in a particular order....words must be so arranged that the ideas will follow one another in their natural order, and the words in their natural connection...."

Stall in his monumental work *Word Order in Sanskrit and Universal Grammar* (1967) notes the distinction between *sambandha* 'the relation of one word to another within a sentence' and *abhisambandha, ānupūrvya* or *ānupūrvī* 'the order or arrangement of words'. He explains that Sanskrit grammarians were interested in *sambandha* or grammatical relations and not in *abhisambandha* or word order.

Thus, in Sanskrit the relation of words is shown by their grammatical terminations. Sentences like *damanakaḥ gacchati* and *gacchati damanakaḥ*, which differ only in the arrangement of the words, are considered synonymous. The differences in such instances are regarded as due to emphasis or stylistic effect and without grammatical significance. The computational program has to learn the inflections of the words and based on this information has to find out what function a word has in a sentence.

3.1.6. Homophony of Pronouns and Conjunctions

As is shown in 2.1.4., some third person pronoun, relative and demonstrative forms are homophonous with some conjunctions in the language. These forms include *tasmāt, tat,*

yat, and *yena* which have been used in the text in question. These words serve as linkers and they join the preceding sentence/clause to the following sentence. Their position in a sentence is also fixed as they invariably occur in between two sentences. Their syntactic position cannot be changed. Though we are concerned only in resolving the lexical anaphors and they happen to be not homophonous with any of these given linkers, it is important here to note that when we would be designing a holistic system, we would have to keep in mind this homophony. Also, if the anaphora resolution system is completely dependent upon the POS tagging of the words, like in the current approach, the POS tagging also needs to be done very carefully, as the entire burden is now on POS tags. The system be it POS tagger or anaphora resolution system, must be able to identify the linkers and the pronominals.

These linkers frequently occur in the discourse, as is evident from PT. They serve as clause markers too. The following examples given here try to show the above contrast in their use. In (3) *tasmat* is functioning like a linker and links the preceding verse with the following sentence. It connects the relation of cause and effect; the utterance before *tasmāt* constitutes the cause and the one after *tasmāt* is the effect:

(3) *yā* *punaḥ* *tri-stanī* *kanyā* *yā-ti* *locana-gocaram.*
REL.F again three-breast_with daughter go-3SG.PRS eye-visible

pitara-m *nāśa-yati* *eva* *sā* *drutam, na atra saṃsayaḥ.*
father.ACC destroy-3SG.PRS emph she quickly not here doubt

***tasmāt* asyāḥ** *darśanam* *pariharatu* *devaḥ.*
SO 3SG.PROX.F.GEN appearance remove.IMP.3SG lord

'And, the daughter with three breasts when passes through the eyes of her father, definitely, destroys him quickly and there is no doubt in it. So, Lord, remove its appearance.' <andʰakakubjakatristanī-katʰā, apkr>

In the following sentence *tasmāt* and *yena* both has been used to join different pairs of sentences.

65

(4)   *mat-pṛṣtʰa-gataḥ*   *tila-mātram*   *api*   *cali-tum na   śakno-ti*

1POSS.SG-back-go.PSPL sesame-quantity EMPH go-GRN   not can-3SG.PRS

*tasmāt katʰa-yāmi asya   nija-abʰiprāyam   **yena**   abʰīṣta-devatā-*

so   say-1SG.PRS 3SG.PRX.GEN self-intention.ACC so_that dear-god-

smaraṇa-m   karoti.

remembrance-ACC   does

'(this monkey) seated on my back cannot move even for an inch, so, I say my

intention to him, so that he prays his desired god.' <prastāvanā-katʰā, ldpr>

In (5), (6), and (7) the preceding utterances have not been given as the sense is clear.

(5)   ***tat**   saṃkṣepamātram śāstram   kiɲcit etesām   prabodʰana-artʰam cintyatām*

then brief-only   discipline some 3SG.PL.GEN awakening-for   think

iti

QUOT

'...then, think of some brief discipline for awakening these children.'
<katʰāmukʰam, mtbd>

(6)   ***tat**   enam   māyā-vacan-aiḥ   viśvās-ya   aham   cʰātra-tām*

then 3SG.PRX.ACC false-word.PL.INS convince-GRN 1SG.NOM student-ness.ACC

*vrajāmi   **yena**   saḥ viśvas-taḥ   kadācit mama   hastagataḥ bʰaviṣyati.*

go.1SG.PRS so_that he   convince-PSPL perhaps 1SG.GEN hand-came be.FUT.3SG

'...then, convincing him by false words, I become his disciple, so that he would

be convinced, perhaps, will be in my hands.' <dūtijambūkāṣādʰabʰūti-katʰā,

mtbd>

(7)   ***yat** prabʰūt-e   api   vitt-e   artʰopāyāḥ   cintanīyāḥ   kartavyāḥ*

that enormous-LOC EMPH wealth-LOC wealth-means thinkable.PL doable.PL

ca   iti

and QUOT

'.....that, even after having a lot of money, the solutions for earning money is thinkable and doable.' <prastāvanākatʰā, mtbd>

The sentences in (103), (104), and (105) contain the pronominal counterparts of the above linkers. This evidence is taken from the text only.

(8) *api svalpa-tara-m kārya-m yat bʰav-et prtʰivī-pateḥ*

EMPH little-very-ACC work-ACC REL.SG.NOM be-3SG.POT earth-husband.GEN

tat na vācyam sabʰā-madʰy-e provāca idam

3SG.N.DST.NOM not speak-GRN assembly-middle-LOC said DEM.N.PRX.ACC

brhaspatiḥ.

brhaspati-NOM

'Even if there is a little bit of work of a king, that must not be spoken in an assembly. This was said by the scholar Brhaspati.' <kīlotpāṭivānara-katʰā, mtbd>

(9) *saḥ api tat ākarnya āragʰaṭṭagʰaṭikām āśritya*

3SG.NOM EMPH DEM.SG.ACC hear.GRN well_water_jar.F.ACC shelter.GRN

tasmāt kūp-āt viniṣkrāntaḥ.

3SG.DEF.ABL well-SG.ABL came_out

'After hearing that (message), he also having taken the shelter in the water jar of the well came out from that well.' <gaṅgadattapriyadarśanayoḥ-katʰā, ldpr>

(10) *māma! asti kaścit upāya-ḥ yena asmākam rakṣā*

uncle.VOC! is any solution-NOM REL.SG.INS 1PL.GEN protection.NOM

bhava-ti?

be-3SG.PRS

'Uncle, is there any way out for our protection?' <bakakarkaṭaka-katʰā, mtbd>

3.1.7. Zero Pronouns

Sanskrit is a pro-drop language. There is a strong agreement between the subject and the verb of a sentence. Sanskrit verb encodes the person and number features of its subject, and due to which the subject is often dropped, as the information regarding the subject can be sought from the verb itself. In the text of PT it has been well-observed that zero pronouns, i.e. pro-drops, occur frequently. The pro-dropping is a practical issue for translation students when rendering Sanskrit texts into English or English like languages. When such constructions are translated in other languages this zero pronoun has to be tracked down, because the system of the target language may not allow this pro-dropping. Such cases have to be resolved in the text relying entirely upon the verb that will tell us person and number features of the dropped pronoun, as is exemplified in the following sentences:

(11) *tat śrutvā piṅgalakaḥ cintayāmāsa-*

  that hear-GRN Pingalak thought

  "yogyaḥ ayam dṛśyate tat katʰa-yāmi etasya agre ātmanaḥ

  worthy this looks then say-1SG.PRS 3SG.PRX.GEN before self.gen

  abʰiprāya-m".

  intention-ACC

  'After hearing that, Pingalak thought- "This seems to be a worthy person so I tell my intention to him".' < kīlotpāṭi-vānarakatʰā, mtbd>

In (11) the first person singular has been dropped, due to the number and person encoding in the verb. In English translation this dropped pronoun has been supplied. In the following sentence (12) either monkey or third person singular pronoun has been dropped, however, this information can be retrieved from the verb.

(12) *mat-pṛṣṭʰa-gataḥ tila-mātram api cali-tum na śakno-ti*

  1POSS.SG-back-go.PSPL sesame-quantity EMPH go-GRN not can-3SG.PRS

tasmāt kaṭʰa-yāmi   asya   nija-abʰiprāya-m   yena   abʰīṣṭa-devatā-

so   say-1SG.PRS 3SG.PRX.GEN self-intention-ACC so_that dear-god-

smaraṇam   karoti.

remembrance does

'(this monkey) seated on my back cannot move even for an inch, so, I say my

intention to him, so that he prays his desired god.' <prastāvanā-katʰā, ldpr>

These cases of zero pronouns require special treatment for their automatic resolution. The system has to be enabled to read the grammatical features of verbs in case it does not find a suitable pronoun for the antecedent of an anaphor. However, our system does not handle zero pronouns.

3.2. SARS Design

3.2.1. Overview of the System Architecture

The working of the system is as follows: At present state, the input to the system is POS tagged data of Sanskrit in which case, number, gender, person etc. are given. The grammatical features given through these tags are, in fact, main clues in identifying the antecedents of anaphors. The system considers one sentence at a time. First, it checks whether a reflexive or reciprocal is available in the sentence in question. If they are not there then it leaves that sentence and considers next sentence for the same. And if they are there, it searches for its antecedents as per our rules (see the algorithm in 3.2.3.). The candidates for being the antecedents of reflexives (PRF) and reciprocals (PRC) are common nouns (NC), proper nouns (NP), personal pronouns (PPR) and relative pronouns (PRL). All these categories are simply identified on the basis of their morphosyntactic tags. Thus POS tagging information is very crucial for anaphora resolution in this system.

Though the experiment was carried out on the POS tagged text of PT, the system could be used for other Sanskrit texts as well if they are POS tagged like PT. The pre-processing of these texts will involve the external sandhi-splitting and POS tagging. The Sanskrit language uses heavy compounding. Even pronouns are compounded with some

element. The compounding of pronouns needs to be analysed and tagged in order to be consistent with the rest of the data. And this will require a separate module called Compound Processor. However, SARS does not handle the compounding of pronouns, but in future this would have to be included in the system.

3.2.2. The Annotated Information

Unlike other anaphora resolution approaches, this system does not require a parser for the pre-processing of the input text. It only needs a POS tagger which tags the Sanskrit text with major grammatical categories along with morpho-syntactic attributes. By tagging information the system will know which anaphors are and which possible antecedents are. The determinants of the exact antecedents are morpho-syntactic features of the words in our approach. As I have stated earlier, the data of the PT was tagged manually with MSRI tagset and currently is the input for our system, but in order to enable the system to be used for other texts also we need to develop a POS tagger for automatic POS tagging of the desired texts, as manual tagging is very costly and time taking. This work was a small experiment in handling lexical anaphors with this approach. But for a long run of the system we would have to modularize the system and develop those modules (like sandhi splitter, compound processor and POS tagger) separately and efficiently. In fact, the modules required for the development of POS tagger, are indirectly required for the development of this system as well.

3.2.3. The SARS Algorithm

The system considers only lexical anaphors (reflexives and reciprocals) and it identifies intrasentential antecedents of these anaphors in the input text. The algorithm is based on our study of lexical anaphors in the Chapter 2 of this dissertation. The input to the anaphora resolution system is the POS tagged (with MSRI Indic language tagset) text.

3.2.3.1. *Algorithm for Reflexive Anaphora*

1. Tokenize each sentence (S) of the input text.
2. Pick up the S in which PRF tag occurs.

3. Check whether the S has NP, NC, PPR, or PRL.

4. Consider all the NP, NC, PPR, and PRL in the S that precede the PRF.

5. Check whether NP, NC, PPR, or PRL has/have .nom tag.

6. If one of the NP, NC, PPR, or PRL has .nom tag, then that word is identified as the antecedent of the PRF.

7. If more than one of these have .nom tag then the nearest to PRF would be its antecedent.

8. If the conditions 6 or 7 are not met then the NP, NC, PPR, or PRL having .ins tag would be considered the antecedent of the PRF.

3.2.3.2. *Algorithm for Reflexive Cataphora*

9. If the S does not have any preceding NP, NC, PPR, or PRL with .nom tag then consider the following NP, NC, PPR, or PRL containing .nom tag.

10. If one of the following NP, NC, PPR, or PRL has .nom tag, then that word is identified as the antecedent.

11. If more than one of these have .nom tag, then the nearest to the PRF would be its antecedent.

12. If the conditions 9 or 10 are not met then the NP, NC, PPR, or PRL having .ins tag would be considered the antecedent of the PRF.

3.2.3.3. *Algorithm for Reciprocal Anaphora*

1. Tokenize each sentence (S) of the input text.

2. Pick up the S in which the PRC tag occurs.

3. Check whether the S has NP, NC, PPR, or PRL.

4. Consider all the NP, NC, PPR, and PRL in the S that precede the PRC.

5. Check whether NP, NC, PPR, or PRL has/have .du.nom or .pl.nom tag.

6. If one of the NP, NC, PPR, or PRL has .du.nom or .pl.nom tag, then that word is identified as the antecedent of the PRC.

7. If more than one of these have .du.nom or .pl.nom tag then the nearest to the PRC would be its antecedent.

71

8. If all of them are containing .sg.nom tag then all of them would be antecedents of the PRC.

9. If the conditions 6, 7 or 8 are not met then the NP, NC, PPR, or PRL having .du.ins or .pl.ins tag would be considered the antecedent of the PRC.

3.2.3.4. *Algorithm for Reciprocal Cataphora*

10. If the S does not have any preceding NP, NC, PPR, or PRL then consider the following NP, NC, PPR, or PRL containing .du.nom or .pl.nom tag.

11. If one of the following NP, NC, PPR, or PRL has .du.nom or .pl.nom tag, then that word is identified as the antecedent of the PRC.

12. If more than one of the following NP, NC, PPR, or PRL have .du.nom or .pl.nom tag, then the nearest to the PRF would be its antecedent.

13. If all of the following NP, NC, PPR, or PRL have .sg.nom tag, then all of them would be collectively identified as the antecedent of the PRC.

14. If the conditions 11, 12 or 13 are not met then the NP, NC, PPR, or PRL having .du.ins or .pl.ins tag would be considered the antecedent of the PRC.

3.2.4. Testing of SARS

SARS has been tested on 3659 sentences from the POS tagged text of PT. These sentences are of various types that have been given to SARS for anaphora resolution. Before entering the data into the program for resolution we have noted the number of anaphor occurrences in the input text. In the input data 91 PRF (reflexives) and 25 PRC (reciprocals) were found (total 116 anaphor occurrences). Later on the number of anaphor antecedent pairs resolved by the system is noted. The intersentential and zero pronoun cases have also been noted; such cases were 19 for reflexives and no reciprocal has been found in such a situation. All these readings have been taken manually. The number of correctly and incorrectly found anaphor antecedent pairs is noted. Some input-output pairs have been presented in section 4.3. of the following chapter. The Table 9 presents the summary of results we have obtained after testing SARS:

Anaphor type	Total	Correct	Wrong	Unresolved (intersentential)
Reflexives	91	50	22	19
Reciprocals	25	14	11	0
Total	116	64	33	19

Table 9: Evaluation results of Sanskrit anaphors

As we have made it clear in the outset, this system is not handling intersentential anaphors and zero pronoun cases. The unresolved anaphors given in the above table are intersentential ones and those whose antecedents are zero pronouns. Few test outputs are given in 4.3. in the following chapter.
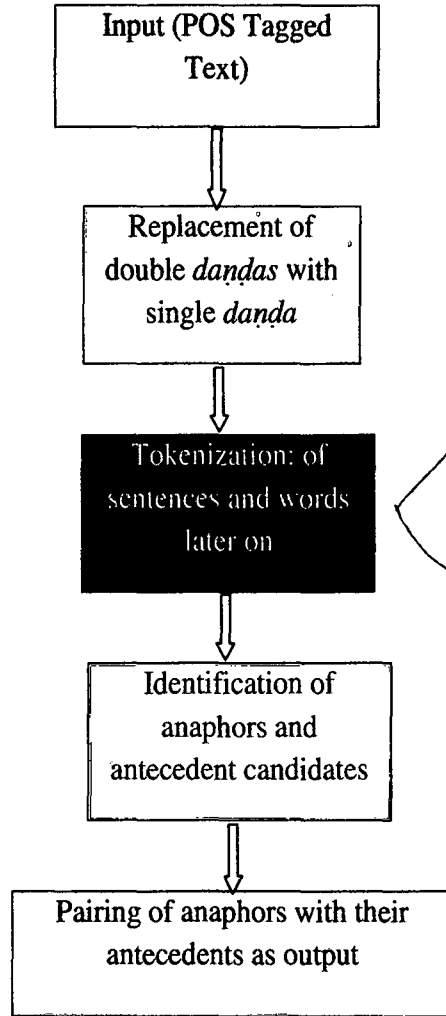
Chapter 4

SARS Implementation

This chapter describes the implementation of the system. SARS is implemented in Java and JSP and runs on PCs on Windows environment. However, it is platform independent and can run on any platform. Though PROLOG and LISP are considered the preferred programming languages for Artificial Intelligence and Natural Language Processing applications, we deliberately choose object oriented methodology of Java for the implementation of the SARS. Java was chosen because of its excellent features of supporting object-oriented concepts. Also it has features of modular design, abstraction, encapsulation, polymorphism, inheritance, portability, reusability and readability. All these features of Java language facilitate the system expandability and versatility in SARS design. Moreover, Java is also preferable to other programming languages from security perspective especially in internet based programs.

4.1. The Model of the System

The SARS has four modules to undergo for processing before it gives any output. These are pre-processor which replaces the double *daṇḍa* with single *daṇḍa* which works as a delimiter for tokenisation; String tokeniser which tokenises sentences and later on words; Tag checker which checks the possible anaphors and antecedents on the basis of tags, and finally Morph checker (morphosyntactic features checker) to determine the resolution.

4.1.1. Model Diagram

Figure 4.1 presents overall functioning of SARS in the given diagram.

```
┌─────────────────────────┐
│     Input (POS Tagged   │
│          Text)          │
└─────────────────────────┘
            ⇓
┌─────────────────────────┐
│     Replacement of      │
│   double dandas with    │
│      single danda       │
└─────────────────────────┘
            ⇓
┌─────────────────────────┐
│    Tokenization: of     │
│   sentences and words   │
│        later on         │
└─────────────────────────┘
            ⇓
┌─────────────────────────┐
│    Identification of    │
│      anaphors and       │
│   antecedent candidates │
└─────────────────────────┘
            ⇓
┌─────────────────────────┐
│ Pairing of anaphors with their │
│   antecedents as output │
└─────────────────────────┘
```

Syntactic tokenization

lexical tokenization

Figure 4.1. The model diagram of SARS

4.1.2. Brief Description of the Processes

The input for the system is POS tagged data of Sanskrit. It goes through five processes before it gets the final result as is shown in the diagram. The input is pasted in the text area of the system which is there in the homepage of SARS, and then is sent for processing and resolution. The pre-processor of the system replaces the double *dandas* with single *dandas* (the double *dandas* are used in the text at the end of a verse as we have discussed it in 3.1.1). Afterwards, the data is tokenized in single sentences delimited by a single *danda*. After sentence tokenization each word is tokenized. The system considers a single tokenised sentence at a time for resolving the anaphors. In each

75

tokenized sentence the system first checks whether a reflexive or reciprocal is available in the sentence in question. If they are not there then it leaves that sentence and considers next sentence for the same. And if they are there, it searches for its antecedents as per our rules (see the algorithm in 3.2.3.) and resolves it. After resolving the anaphor in a sentence it moves on to the next sentence and operates likewise till the end of the string. The candidates for being the antecedents of reflexives (PRF) and reciprocals (PRC) are common nouns (NC), proper nouns (NP), personal pronouns (PPR) and relative pronouns (PRL) as per our generalisations in the PT. All these categories are simply identified on the basis of their morphosyntactic tags. Thus POS tagging information is very crucial for anaphora resolution in this system.

4.2. Implementation Strategies

The program has been prepared on a Windows platform using tools and methods described below. This program, however, is platform independent and can run on any platform.

4.2.1. An Overview of the Methods and Techniques Used

SARS is planned to develop in web architecture. The system tools are Java based. Front end is done in Java Server Pages (JSP) running on the web server called Apache Tomcat. The programming has been done in the Java environment. The system accepts the Sanskrit text in Devanagari UTF-8 format and returns output in similar format. The system takes the input in text area, and then sends it for tokenisation and other steps described in 4.1.2. The following is an overview of the methods and techniques used in developing the program.

4.2.1.1. *Front End*

The front end of the program is Java Server Pages file named *request.jsp*. The front end opens in a web browser that is based locally on the user's computer. The URL opens the JSP file located on the host computer usually at the path <C:\Program Files\Apache Tomcat 4.0\webapps\sars>. The browser, with the help of the java-web server, reads the

request.jsp file. The page contains codes of HTML, JSP and Java languages. This page sends the input data to *result.jsp* page for further processing. To understand the structure of the jsp file, we need to know the following preliminary things.

4.2.1.1.1. *Java Server Pages (JSP)*

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. It is one of the most sophisticated tools available for high performance and secures web applications.

4.2.1.1.2. *Java*

Java is an object-oriented programming language developed by a team led by James Gosling at Sun Microsystems. Originally called *Oak*, it was designed in 1991 for use in embedded consumer electronic appliances. In 1995, renamed *Java*, it was redesigned for developing Internet applications. It was intended to replace C++, although the feature set better resembles that of object C. Sun Microsystems currently maintains and updates Java regularly. Java is a network-oriented programming language that is specifically designed for writing programs that can be safely downloaded to user's computer through the internet and immediately run without fear of viruses or other harm to the computer or files. Using small Java programs (called "Applets"), web pages can include functions such as animation, calculators, and other fancy tricks. Java programs are of three kinds: stand-alone executable programs, applets, and servlets.

4.2.1.1.3. *Applets*

An applet is a small program written in Java and embedded in a HTML page using the tag <APPLET>. It safely downloads on the client machine and runs using the client Java runtime environment. Applets differ from full-fledged Java applications in that they are not allowed to access certain resources on the local computer, such as files and serial devices (modems, printers, etc.), and are prohibited from communicating with most other

computers across a network. The current rule is that an applet can only make an internet connection to the computer from which the applet was sent.

4.2.1.1.4. Servlets

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.

4.2.1.1.5. HTML

HTML or Hyper Text Mark-up Language is a markup language: a simple language for laying out documents, linking documents on the Internet, and bringing images, sound, and video alive on the Web. However, it cannot interact with the user except through simple forms. Web pages in HTML are essentially static and flat. It is the oldest language that served the purpose of developing web pages.

4.2.1.2. Java Objects

The JSP file called *result.jsp* uses a java object called STokeniser which utilises the services of Pre-processor, that is, Tokens. These .class files are the compiled programs written in Java that tokenize the input text. The Pre-processor object first replaces the double *dandas* with single *dandas* and then the STokeniser tokenises each sentence in the input data with the help of delimiter single *dandas*. There is another Java object called checkCoRef that processes the tokenized text and performs the resolution. There are several modules in this program that take care of several processes undergoing during the mechanical analysis. Below is given the description of the modules involved in the program. Wherever necessary, important snippets of the code written in Java have been presented.

78

4.2.1.2.1. Pre-processor

As is illustrated in the diagram above, the input first goes to the pre-processor module and checks the presence of double *daṇḍas*. It replaces these unwanted double *daṇḍas* with single *daṇḍas*. To do this is necessary because for tokenisation we use the delimiter single *daṇḍas* "।". Though not always, but most often we get complete sentences in the tokenized units. For example if the input is given the following verse,

किम्\CNT तया\PPR.fem.sg.3.ins.iii.n.n.dst क्रियते\V.apd.sg.3.prs.n धे न्\NC.fem.sg.ins.iii या\PRL.fem.sg.3.0.i न\CNG सू षि\V.ppd.sg.3.prs.n न\CNG दु ग्ध्वा\JJ.fem.sg.0.i.n.n ।\PU कः\PWH.mas.sg.3.nom.i अर्थ \NC.mas.sg.nom.i पु त्रे\NC.mas.sg.ins.iii जाते �$KDP.mas.sg.ins.iii यः\PRL.mas.sg.3.nom.i न\CNG विद्वान्\JJ.mas.sg.nom.i.n.n न\CNG भक्तिमान्\JJ.mas.sg.nom.i.n.n ॥\PU 6\RDS ॥\PU

After tokenization it will look like,

किम्\CNT तया\PPR.fem.sg.3.ins.iii.n.n.dst क्रियते\V.apd.sg.3.prs.n धे न्\NC.fem.sg.ins.iii या\PRL.fem.sg.3.0.i न\CNG सू षि\V.ppd.sg.3.prs.n न\CNG दु ग्ध्वा\JJ.fem.sg.0.i.n.n ।\PU कः\PWH.mas.sg.3.nom.i अर्थ \NC.mas.sg.nom.i पु त्रे\NC.mas.sg.ins.iii जाते �$KDP.mas.sg.ins.iii यः\PRL.mas.sg.3.nom.i न\CNG विद्वान्\JJ.mas.sg.nom.i.n.n न\CNG भक्तिमान्\JJ.mas.sg.nom.i.n.n ॥\PU 6\RDS ।\PU

The replacement is operated on the second double *daṇḍas* like herein the double *daṇḍa* after "6\RDS" has been replaced not the first one. It happens because the program does not replace a double *daṇḍa* if it is followed by a number like "6\RDS" in this case.

Code snippet

```
public String repString(){
        int count=0;

        while(this_text!=null){
                if (this_text.indexOf(endLine)>-1){
```

```
            System.out.println("Sankhya.....");
            count=count+1;
            if(count==1){
                    System.out.println("Sankhya.....1");

    p1=this_text.substring(0,this_text.indexOf(endLine)).trim();
                    rtext=rtext+p1+" "+endLine;

    p2=this_text.substring(this_text.indexOf(endLine)+endLine.length(),this_text.len
gth()).trim();
                    this_text=p2;
            }
            else{
                    System.out.println("Sankhya.....2");

    p1=this_text.substring(0,this_text.indexOf(endLine)).trim();
                    rtext=rtext+p1+" "+endLine1;

    p2=this_text.substring(this_text.indexOf(endLine)+endLine.length(),this_text.len
gth()).trim();
                    this_text=p2;
                    count=0;

            }
    }
    else{
            System.out.println("Sankhya.....3");
            rtext=rtext+this_text;
            this_text=null;

    }
    }

    return rtext;
    }
}
```

4.2.1.2.2. *String Tokeniser*

The Java object called STokenizer tokenises each sentence delimited by single *danda* and later on each word delimited by space. For anaphora resolution the recognition of a sentence is mandatory in our approach, as SARS handles only intrasentential anaphors. After sentence tokenisation the module performs word tokenisation.

Code snippet

```java
public class STokenizer{

        Hashtable h = null;
        int c = 0;
        String fs = "";
        String rs ="";
        int iter = 0;

        public STokenizer(String s, String del){
                h = new Hashtable();
                iter=0;
                c=0;
                rs="";
                fs ="";

                if (s.indexOf(del)==-1){
                        //System.out.println("string does not have VP");
                        h.put(new Integer(0) , s);
                }
```

4.2.1.2.3. Reference Resolution

The Java object checkCoRef which can be conveniently called reference resolution does
the main work of this enterprise. It searches for reflexives and reciprocals with the help of
POS tags given in the input data and picks up those sentences only. Following the given
conditions in the algorithm this identifies the antecedent for a particular anaphor. It
discards the sentences with PRC or PRF which do not have the antecedents.

Code snippet

```java
public String coRefProcess(String pShabd){
                String word="",out="";
                if(gVaakye.indexOf(pShabd)>-1)
                  {
                        lVaakye=gVaakye.substring(0,gVaakye.indexOf(pShabd)).trim();
                  }
                else
                  {
                        lVaakye="";
                  }

        rVaakye=gVaakye.substring(gVaakye.indexOf(pShabd)+pShabd.length(),gVaaky
e.length()).trim();
```

```
if(lVaakye.indexOf("NP")>-1 || lVaakye.indexOf("NC")>-1 ||
lVaakye.indexOf("PPR")>-1 || lVaakye.indexOf("PRL")>-1){
            if (lVaakye.indexOf (".nom")>-1){
                    st1=new StringTokenizer(lVaakye," ");
                    while(st1.hasMoreTokens()){
                            word=st1.nextToken();
                            if(word.indexOf (".nom")>-1){
                                    out=word;
                            }
                    }
            }
}
            else if(rVaakye.indexOf("NP")>-1 || rVaakye.indexOf("NC")>-1 ||
rVaakye.indexOf("PPR")>-1 || rVaakye.indexOf("PRL")>-1){
            if (rVaakye.indexOf (".nom")>-1){
                    st1=new StringTokenizer(rVaakye," ");
                    while(st1.hasMoreTokens()){
                            word=st1.nextToken();
                            if(word.indexOf (".nom")>-1){
                                    out=word;
                            }
                    }
            }
}
```

4.2.1.3. Web server: Apache Tomcat 4.0

For the web server we have used Apache – Tomcat technology. Apache-Tomcat is a web server software to develop and run Java Server Pages on a host. The Apache-Tomcat, earlier Tomcat, was developed by the Apache association of developers. It is intended to be a collaboration of the best-of-breed developers from around the world. There are several versions of Tomcat available freely on the web. We have used the version 4.0. Tomcat 4.0 implements the servlets and Java server pages specifications from Java Software, and includes many additional features that make it a useful platform for developing and deploying web applications and web services.

4.3. Running of SARS: An Evaluation of the Results

The URL of the application is http://sanskrit.jnu.ac.in/sars The home page of the site looks like–
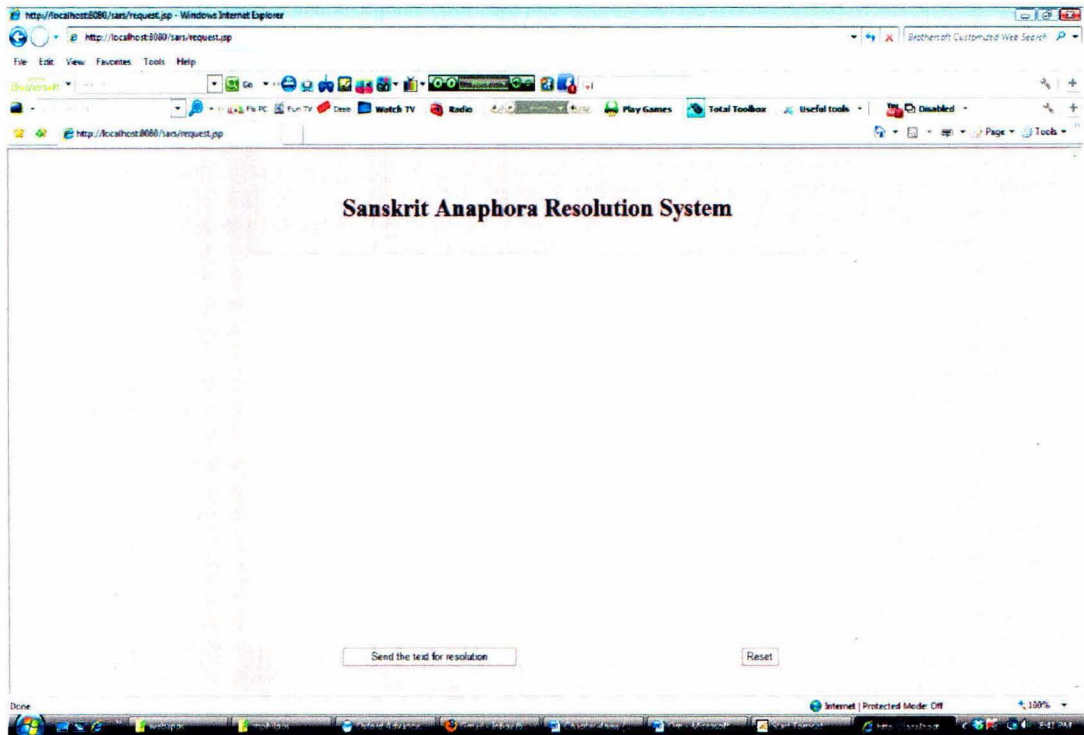
82

Figure 4.2. Home Page of SARS

The user will paste the POS tagged Sanskrit data in the text area of the system shown in the above figure. After inputting the data user will click on the "Send the text for resolution" button and the present page will send the data to *result.jsp* page for processing. The result on the system will look like the following figure in the next page.
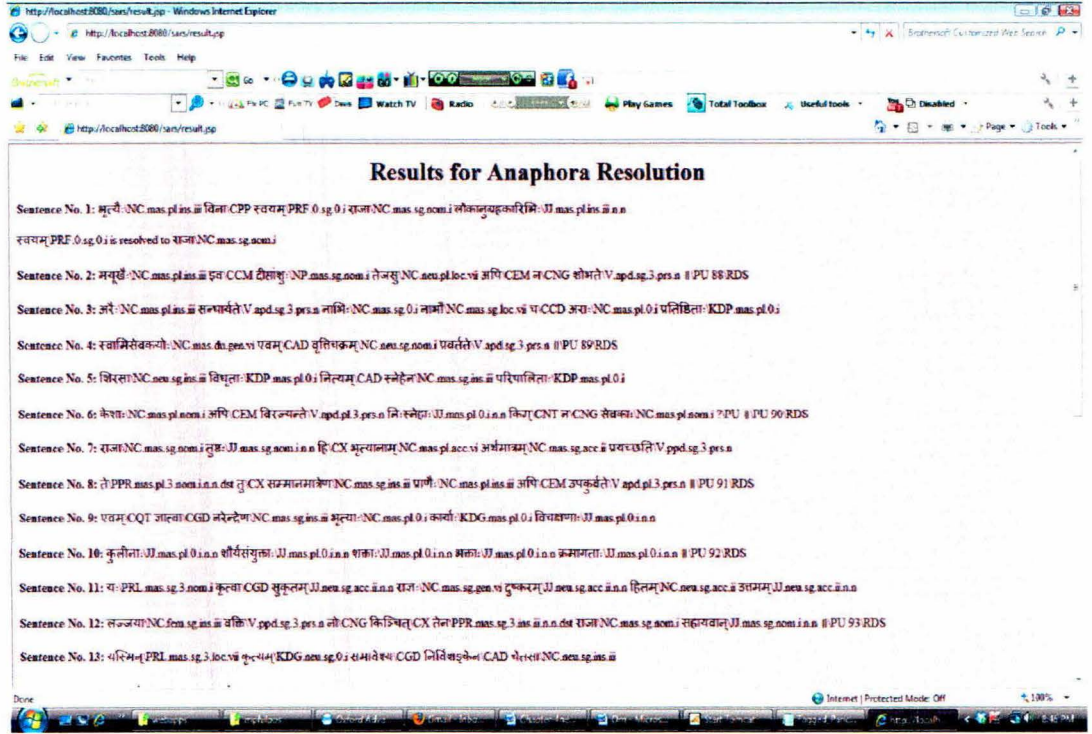
Figure 4.3. Results for Anaphora Resolution by SARS

Some Input-Output Examples

1. Input: स:\PPR.mas.sg.3.nom.i.n.n.dst च\CCD तम्\PPR.mas.sg.3.acc.ii.n.n.dst परिज्ञाय\CGD समीपम्\JJ.neu.sg.0.i.n.n गत्वा\CGD स्वयम्\PRF.0.sg.0.i एव\CEM कौतु काष्ट\NC.neu.sg.abl.v अताडयत्\V.ppd.sg.3.imPRF.n I\PU

Output: Sentence No. 1: स:\PPR.mas.sg.3.nom.i.n.n.dst च\CCD तम्\PPR.mas.sg.3.acc.ii.n.n.dst परिज्ञाय\CGD समीपम्\JJ.neu.sg.0.i.n.n गत्वा\CGD स्वयम्\PRF.0.sg.0.i एव\CEM कौतु काष्ट\NC.neu.sg.abl.v अताडयत्\V.ppd.sg.3.imPRF.n

स्वयम्\PRF.0.sg.0.i is resolved to स:\PPR.mas.sg.3.nom.i.n.n.dst

2. Input: तदनु\CAD सा\DAB.fem.sg.3.nom.i.dst.n नापिती\NP.fem.sg.nom.i ताम्\DAB.fem.sg.3.acc.ii.dst.n स्वसखीम्\NC.fem.sg.acc.ii बन्धनात्\NC.neu.sg.abl.v विमोच्य\CGD तस्या:\PPR.fem.sg.3.gen.vi.n.n.dst स्थाने\NC.neu.sg.loc.vii यथापू र्व\CAD आत्मानम्\PRF.mas.sg.acc.ii बद्ध्वा\CGD ,\PU ताम्\PPR.fem.sg.3.acc.ii.n.n.dst दे वदत्तसकाशे\NC.mas.sg.loc.vii सङ्के तस्थानम्\NC.neu.sg.acc.ii प्रे षितवती\KDP.fem.sg.0.i I\PU

84

Output: Sentence No. 1: तदनु\CAD सा\DAB.fem.sg.3.nom.i.dst.n
नापिती\NP.fem.sg.nom.i ताम्\DAB.fem.sg.3.acc.ii.dst.n स्वसखीम्\NC.fem.sg.acc.ii
वन्धनात्\NC.neu.sg.abl.v विमोच्य\CGD तस्याः\PPR.fem.sg.3.gen.vi.n.n.dst
स्थाने\NC.neu.sg.loc.vii यथापूर्वम्\CAD आत्मानम्\PRF.mas.sg.acc.ii वद्ध्वा\CGD ,\PU
ताम्\PPR.fem.sg.3.acc.ii.n.n.dst देवदत्तसकाशे\NC.mas.sg.loc.vii
सङ्केतस्थानम्\NC.neu.sg.acc.ii प्रेषितवती\KDP.fem.sg.0.i

आत्मानम्\PRF.mas.sg.acc.ii is resolved to नापिती\NP.fem.sg.nom.i

Some other test outputs are given in the following sentences with their sentence IDs
produced by the system:

Sentence No. 135: यः\PRL.mas.sg.3.nom.i न\CNG आत्मना\PRF.mas.sg.ins.iii न\CNG
च\CCD परेण\PPR.mas.sg.3.ins.iii.n.n.dst च\CCD बन्धुवर्गो\NC.mas.sg.loc.vii ,\PU
दीने\NC.mas.sg.loc.vii दयाम्\NC.fem.sg.acc.ii न\CNG कुरुते\V.apd.sg.3.prs.n न\CNG
च\CCD भृत्यवर्गो\NC.mas.sg.loc.vii

आत्मना\PRF.mas.sg.ins.iii is resolved to यः\PRL.mas.sg.3.nom.i

Sentence No. 140: आरोहति\V.ppd.sg.3.prs.n न\CNG यः\PRL.mas.sg.3.nom.i
स्वस्य\PRF.mas.sg.gen.vi वंशस्य\NC.mas.sg.gen.vi अग्रे\CAD ध्वजः\NC.mas.sg.0.i
यथा\CCM ॥\PU 27\RDS

स्वस्य\PRF.mas.sg.gen.vi is resolved to यः\PRL.mas.sg.3.nom.i

Sentence No. 3345: इदानीम्\CX त्वम्\PPR.0.sg.2.nom.i.n.n.0 आत्मनः\PRF.mas.sg.gen.vi
अन्यत्\DAB.neu.sg.3.acc.ii.0.n बाहुयुगलम्\NC.neu.sg.acc.ii द्वितीयम्\JQ.neu.sg.acc.ii.ord.n
शिरः\NC.neu.sg.acc.ii च\CCD याचस्व\V.apd.sg.2.imp.n ,\PU येन\CSB
पट्ट्रयम्\NC.neu.sg.acc.ii संपादयसि\V.ppd.sg.2.prs.n पुरतः\CX पृष्ठतः\CX च\CCD

आत्मनः\PRF.mas.sg.gen.vi is resolved to त्वम्\PPR.0.sg.2.nom.i.n.n.0

Sentence No. 433: तत्\PPR.neu.sg.3.acc.ii.n.n.dst श्रुत्वा\CGD गतायुषम्\JJ.neu.sg.0.i.n.n
इव\CCM आत्मानम्\PRF.mas.sg.acc.ii मन्यमानः\KDP.mas.sg.0.i सञ्जीवकः\NP.mas.sg.nom.i
परम्\JQ.neu.sg.acc.ii.nnm.n विषादम्\NC.mas.sg.acc.ii अगमत्\V.ppd.sg.3.aor.n

आत्मानम्\PRF.mas.sg.acc.ii is resolved to सञ्जीवकः\NP.mas.sg.nom.i

Sentence No. 582: 170\RDS ॥\PU ततः\CAD स्वामिप्रसादरहितौ\JJ.mas.du.nom.i.n.n
क्षुत्क्षामकण्ठौ\JJ.mas.du.nom.i.n.n परस्परम्\PRC.neu.sg.0.i करटकदमनकौ\NP.mas.du.nom.i
मन्त्रयेते\V.apd.du.3.prs.n

85

परस्परम्\PRC.neu.sg.0.i is resolved to करटकदमनकौ\NP.mas.du.nom.i

Above sentences were handled correctly. Following are some examples where system has given wrong results:

Sentence No. 194: आत्मानम्\PRF.mas.sg.acc.ii किम्\CNT स:\PPR.mas.sg.3.nom.i.n.n.dst न\CNG द्रेष्टि\V.ppd.sg.3.prs.n सेव्यासेव्यम्\JJ.neu.sg.acc.ii.n.n न\CNG वेत्ति\V.ppd.sg.3.prs.n य:\PRL.mas.sg.3.nom.i

आत्मानम्\PRF.mas.sg.acc.ii is resolved to य:\PRL.mas.sg.3.nom.i

Sentence No. 257: तथा\CCM वयम्\PPR.0.pl.1.nom.i.n.n.prx देवपादानाम्\NC.mas.pl.gen.vi अन्वयागता:\JJ.mas.pl.0.i.n.n भृत्या:\NC.mas.pl.0.i आपत्सु\NC.mas.pl.loc.vii अपि\CEM पृष्ठगामिन:\JJ.mas.pl.0.i.n.n यद्यपि\CSB स्वम्\PRF.mas.sg.acc.ii अधिकारम्\NC.mas.sg.acc.ii न\CNG लभामहे\V.apd.pl.1.prs.n ,\PU तथापि\CSB देवपादानाम्\NC.mas.pl.gen.vi एतत्\PPR.neu.sg.3.nom.i.n.n.prx युक्तम्\JJ.neu.sg.0.i.n.n न\CNG भवति\V.ppd.sg.3.prs.n

स्वम्\PRF.mas.sg.acc.ii is resolved to एतत्\PPR.neu.sg.3.nom.i.n.n.prx

In sentence no. 194 the correct antecedent is स:\PPR.mas.sg.3.nom.i.n.n.dst which was not decided by the system. Likewise in sentence no. 257 the antecedent was वयम्\PPR.0.pl.1.nom.i.n.n.prx but the system instead wrongly chose एतत्\PPR.neu.sg.3.nom.i.n.n.prx.

4.3.1. Result Analysis and Limitation

Currently this system is giving results for intrasentential lexical anaphors. As is described in 3.2.4., it is giving encouraging results. There were 97 intrasentential lexical anaphors and cataphors in the input text and out of which 64 were correctly resolved. However, an improvement is needed for better results. The system is not able to handle intersentential anaphors at present state. Also, if a sentence has both a reflexive and a reciprocal, it picks up one of them and gives the result. The cases of reflexive compounding and reciprocal compounding are not handled. We shall try in future to give better results.

Chapter 5

Conclusion and Further Enhancement

Anaphora resolution has proven to be a formidable task for computers to perform. It plays a significant role in most of the NLP applications such as machine translation (MT), text summarization, question answering systems, information extractions etc. In Sanskrit, however, it has received very little computational research. Until now, only a few works are available on this topic (see 5.1). Anaphora resolution is a very wide topic in NLP and generally, it includes lexical anaphors, third person pronouns, and other anaphoric items.

In this study we have investigated the pronouns and lexical anaphors usage in the Sanskrit text *Pañcatantra*. Also, we have developed a program to identify lexical anaphors and their antecedents automatically. The system developed in the present research has been named SARS (acronym of *Sanskrit Anaphora Resolution System*) and at present it takes care of lexical anaphors only, as this work was a time bound one year (M. Phil.) project, the handling of all the anaphoric cases was not feasible. The system identifies intrasentential antecedents of lexical anaphors in the input text. The antecedent of lexical anaphor could be a common noun, proper noun, personal pronoun or relative pronoun. They (antecedents) occur either in nominative or instrumental or very rarely in genitive casal forms, as we have observed in our investigation of the text. The system has been tested in a corpus of 3659 sentences of PT containing 116 anaphors and is giving encouraging results.

The work done in this dissertation can broadly be classified in three parts:
• Data collection from the text
• Analysis of the data
• Development of the system

The method of data collection, transcription and formatting has been described in the introductory chapter. The second chapter gives a complete description of pronouns and lexical anaphor usage in the text. The analysis done in this chapter has been used in developing the algorithm for automatic anaphora resolution. The third chapter discusses the issues and challenges in designing the system. It also presents an algorithm for lexical anaphors resolution. In the fourth chapter, the computing task has been taken up. This chapter describes the methods and techniques applied to bring about the system. Computing has been the final and ultimate goal of the work done here. The system architecture and design is explained with the sample code of JSP and Java. Finally, the implemented system's functioning is given with development information and screenshots.

The work done on Sanskrit anaphora and cataphora resolution through this dissertation is a step forward in the existing research published on this topic. I have tried my best to devise the algorithm for lexical anaphors to suit the usage in the text of study. The classification of pronouns and lexical anaphors in Chapter 2 of this dissertation gives a complete picture of such usage in the text. It may provide a good understanding of these usages in the language for further research on this subject. This study could be tested for the language in general and after investigating some other texts of classical Sanskrit new findings can be added.

5.1. Limitations of the System
The system is able to identify only intrasentential lexical anaphors. There are a lot more anaphoric usages in the language that, yet, need to be handled automatically in order to facilitate the language processing tasks. The intersentential cases of reflexives and reciprocals are yet to be resolved. The system does not recognise clause boundaries and does not work for complex sentences validly. The compounding of all kinds of pronouns is a new type of problem in anaphora resolution research which is specific to Sanskrit. Such structures increase the load of the machine and reduce the efficiency of the system. The present system does not take care of compound reflexives and compound reciprocals.

To resolve them it will require a separate module called compound processor. Also, the system does not consider pro drop, that is, zero pronoun cases.

5.2. Potential Application

The applications of anaphora resolution are enormous. SARS program can prove to be a very useful tool for machine translation systems. If the anaphors and their antecedents are known to machine, the machine can take care of them when they are being translated in typologically different languages as there might be some issues of agreements. To fit to these agreements such resolution is unavoidable. As the system developed is highly scalable, it can be easily adapted and extended to suit the needs of other languages as well.
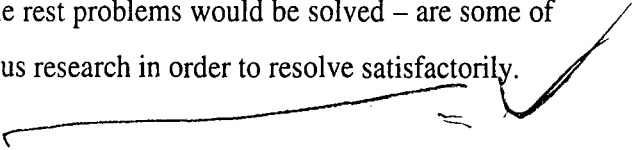
5.3. Scope of Further Research and Development

This is a small effort in designing an anaphor resolution system for Sanskrit. The input to the system is POS tagged data which was tagged with the MSRI Indic Language POS tagset (IL-POST). The linguistic analysis of the data for machine is based on the tagging information only. The system would have to be enhanced to cover wider areas of coreference resolution like intersentential anaphor handling and also to consider third person pronouns at both levels: intrasentential and intersentential. With the help of verbal inflections the cases of zero pronouns can also be handled, as the Sanskrit verb encodes information about the subject also. To use this system professionally for other Sanskrit texts as well we would need the following additions:

- adding a sandhi processing module
- adding lexical resources
- adding compound processing module
- adding a comprehensive POS tagger

For the smooth running of the system the issues discussed in Chapter 3 need to be taken care of. In comparison to other Indian languages Sanskrit texts are a bit complex and therefore their pre-processing has to be done rigorously.

Thus, there is a big scope for research and development in this area of research including the enhancement of this task further. The third person pronouns which are often anaphoric should be investigated for their mechanical handling to smooth the language processing tasks. Their intrasentential and intersentential handling is very necessary for complete anaphora resolution. And also, the use of anaphoric adjectives is a predominant feature of Sanskrit and their resolution is also warranted. To what extent morphological features will solve the problem and how the rest problems would be solved – are some of the relevant questions which require rigorous research in order to resolve satisfactorily.

Appendices

Appendix 1. A Sample of POS Tagged Text

॥\PU श्री:\CIN ॥\PU

पञ्चतन्त्रे\NP.neu.sg.loc.vii अथ\CIN मित्रसम्प्राप्ति:\NP.fem.sg.nom.i

(\PU द्वितीयम्\JQ.neu.sg.nom.i.ord.n तन्त्रम्\NC.neu.sg.nom.i)\PU

अथ\CIN इदम्\PPR.neu.sg.3.nom.i.n.n.prx आरभ्यते\V.apd.sg.3.prs.n

मित्रसम्प्राप्ति:\NP.fem.sg.nom.i नाम\CX द्वितीयम्\JQ.neu.sg.nom.i.ord.n

तन्त्रम्\NC.neu.sg.nom.i ,\PU यस्य\PRL.neu.sg.3.gen.vi

अयम्\DAB.mas.sg.3.nom.i.prx.n आद्य:\JJ.mas.sg.nom.i.n.n श्लोक:\NC.mas.sg.nom.i

-\PU

असाधना:\JJ.mas.pl.nom.i.n.n अपि\CAD प्राज्ञा:\NC.mas.pl.nom.i

बुद्धिमन्त:\NC.mas.pl.nom.i बहुश्रुता:\NC.mas.pl.nom.i ॥\PU

साधयन्ति\V.ppd.pl.3.prs.n आशु\CAD कार्याणि\NC.neu.pl.acc.ii काकाखुमृगकूर्मवत्\CCM

॥\PU 1\RDS ॥\PU

तत्\CX यथा\CCM अनुश्रूयते\V.apd.sg.3.prs.n -\PU अस्ति\V.ppd.sg.3.prs.n

दाक्षिणात्ये\JJ.mas.sg.loc.vii.n.n जनपदे\NC.mas.sg.loc.vii

महिलारोप्यम्\NP.neu.sg.nom.i नाम\CX नगरम्\NC.neu.sg.nom.i ॥\PU

तस्य\PPR.neu.sg.3.gen.vi.n.n.dst न\CNG अतिदूरस्थ:\JJ.mas.sg.0.i.n.n

महोच्छ्रायवान्\JJ.mas.sg.0.i.n.n नानाविहङ्गोपभुक्तफल:\JJ.mas.sg.0.i.n.n

कीटै:\NC.mas.pl.ins.iii आवृतकोटर:\JJ.mas.sg.0.i.n.n

छायाश्वासितपथिकजनसमूह:\JJ.mas.sg.0.i.n.n न्यग्रोधपादप:\NC.mas.sg.0.i

महान्\JQ.mas.sg.0.i.nnm.n ॥\PU अथवा\CCD युक्तम्\JJ.neu.sg.0.i.n.n

उक्तम्\KDP.neu.sg.0.i -\PU

छायासुप्तमृग:\JJ.mas.sg.0.i.n.n शकुन्तनिवहै:\NC.mas.pl.ins.iii

विष्वग्विलुप्तच्छद:\JJ.mas.sg.0.i.n.n ,\PU

कीटै:\NC.mas.pl.ins.iii आवृतकोटर:\JJ.mas.sg.0.i.n.n कपिकुल्यै:\NC.mas.pl.ins.iii

स्कन्धे\NC.mas.sg.loc.vii कृतप्रश्रय:\JJ.mas.sg.0.i.n.n ॥\PU

विश्रब्धम्\JJ.neu.sg.0.i.n.n मधुपैः\NC.mas.pl.ins.iii निपीतकुसुमः\JJ.mas.sg.nom.i.n.n
श्लाघ्यः\JJ.mas.sg.nom.i.n.n सः\DAB.mas.sg.3.nom.i.dst.n एव\CEM
द्रुमः\NC.mas.sg.nom.i ,\PU
सर्वैः\JQ.neu.pl.ins.iii.nnm.n अङ्गैः\NC.neu.pl.ins.iii
बहुसत्त्वसङ्गसुखदः\JJ.mas.sg.nom.i.n.n भूभारभूतः\JJ.mas.sg.nom.i.n.n
अपरः\PPR.mas.sg.3.nom.i.n.n.0 ॥\PU 2\RDS ॥\PU

तत्र\CAD च\CCD लघुपतनकः\NP.mas.sg.nom.i नाम\CX वायसः\NC.mas.sg.nom.i
प्रतिवसति\V.ppd.sg.3.prs.n स्म\CX ॥\PU सः\PPR.mas.sg.3.nom.i.n.n.dst कदाचित्\CX
प्राणयात्रार्थम्\NC.neu.sg.dat.i पुरम्\NC.neu.sg.acc.ii उद्दिश्य\CGD
प्रचलितः\KDP.mas.sg.0.i यावत्\CSB पश्यति\V.ppd.sg.3.prs.n ,\PU तावत्\CSB
जालहस्तः\JJ.mas.sg.nom.i.n.n अतिकृष्णतनुः\JJ.mas.sg.nom.i.n.n ,\PU
स्फुटितचरणः\JJ.mas.sg.nom.i.n.n ,\PU ऊर्ध्वकेशः\JJ.mas.sg.nom.i.n.n
यमकिङ्कराकारः\JJ.mas.sg.nom.i.n.n नरः\NC.mas.sg.nom.i संमुखः\CAD
बभूव\V.ppd.sg.3.prf.n ॥\PU

Appendix 2. The IL-POST

Description of Tags:
(A) Categories and Types (followed by their respective attributes):

(i) Noun (N)
1. Common (NC)
gender, number, case, nominal declension
2. Proper (NP)
gender,number,case, nominal declension

(ii) Verb (V)
pada, number, person, tense\mood, honorificity

(iii) Pronoun (P)
1. Pronominal (PPR)
gender, number, person, case, nominal declension, honorificity, distance
2. Reflexive (PRF)
gender, number, case, nominal declension
3. Reciprocal (PRC)
gender, number, nominal declension
4. Relative (PRL)
gender, number, person, case, nominal declension
5. Wh (PWH)
gender, number, case, nominal declension

(iv) Nominal Modifier (J)
1.   Adjective (JJ)
gender, number, case, nominal declension
2.   Quantifier (JQ)
gender, number, case, nominal declension, numeral

(v) Demonstrative (D)
1. Absolutive (DAB)
gender, number, person, case, nominal declension, distance, emphatic, honorificity
2. Relative (DRL)
gender, number, case, nominal declension
3. Wh- (DWH)
gender, number, case, nominal declension

(vi) Kridant (KD)

      1. Participle (KDP)
      gender, number, case, nominal declension
      2. Gerundive (KDG)
      gender, number, case, nominal declension

(vii) Particle (C)

      1. Coordinating (CCD)
      2. Subordinating (CSB)
      3. Gerundive (CGD)
      4. Interjection (CIN)
      5. Negative (CNG)
      6. Emphatic (CEM)
      7. Interrogative (CNT)
      8. Adverb (CAD)
      9. Postposition (upapada) (CPP)
      10. Quotative (CQT)
      11. Comparative (CCM)
      12. Reduplicative (CRD)
      13. Other (CX)

(viii) Punctuation (PU)

(ix) Residual (RD)

      1. Foreign word (RDF)
      2. Symbol (RDS)
      3. Others (RDX)

(B) Attributes and their values:

1. Gender (Gen)
      a.Masculine (mas)
      b.Feminine (fem)
      c.Neuter (neu)

2. Number (Num)
      a. Singular (sg)
      b. Dual (du)
      c. Plural (pl)

3. Person (Per)
      a. First (1)
      b. Second (2)
      c. Third (3)

4. Case (Cs)
      a.Nominative (nom)
      b.Accusative (acc)
      c.Instrumental (ins)
      d.Dative (dat)
      e.Ablative (abl)
      f.Genetive (gen)
      g.Locative (loc)
      h.Vocative (voc)

5. Nominal declension "*vibhakti*" (Vbh)
      a. Prathama (i)
      b. Dwitiya (ii)
      c. Tritiya (iii)
      d. Chaturthi (iv)
      e. Panchami (v)
      f. Shashthi (vi)
      g. Saptami (vii)
      h. Vocative (viii)

6. Tense/Mood (Tns/Mood)
      a. Present (prs)
      b. Aorist (aor)
      c. Imperfect (imprf)
      d. Perfect (prf)
      e. Periphrastic Future (phf)
      f. General Future (gft)
      g. Imperative (imp)
      h. Potential (pot)
      i. Benedictive (ben)
      j. Conditional (cnd)

7. Numeral (Nml)
      a.Ordinal (ord)
      b.Cardinal (crd)
      c.Non-numeral (nnm)

8. Distance (Dist)
      a. Proximal (prx)
      b. Distal (dst)

9. Emphatic (Emph)
      a. Yes y
      b. No n

10.Negative (Neg)
      a.Yes y
      b.No n

11. Honorificity (Hon)
      a. Yes y
      b. No n

12. Pada (Pd)
      a. Parasmaipada (ppd)
      b. Atmanepada (apd)

Bibliography

Abbi, Anvita (2000), *A Manual of Field Linguistics and Structures of Indian Languages*, Munich: Lincom-Europa.

Ananthanarayana, H.S. (1996), "Relative clause in Indo-Aryan", in Lakshmi, V.S. & Mukherjee, Aditi (eds.), *Word Order in Indian Languages*, pp. 31-45, Hyderabad: Booklinks Corporation.

Apte, Vaman Shivram, (1934), *The Student's Guide to Sanskrit Composition*, Mumbai: The Standard Publishing Company.

Bakharia, Aneesha (2001) *Java Server Page*, New Delhi: Prentice-Hall of India.

Brennan, S.E., M.W. Friedman, & C.J. Pollard (1987), "A Centering Approach to Pronouns", in *Proceedings of the 25th Annual Meeting on Association for Computational Linguistics*, Morristown, NJ, USA, pp. 155-162.

Carter, D. (1987), *Interpreting anaphora in natural language texts*, Chichester: Ellis Horwood.

Chomsky, N. (1981), *Lectures on Government and Binding*, Dedrecht: Foris.

Chandrashekar, R. (2007), *POS tagging for Sanskrit*, unpub. Ph.D. Thesis, New Delhi: Jawaharlal Nehru University.

Carbonell J. & R. Brown (1988), "Anaphora Resolution: A Multistrategy Approach", in *Proceedings of the 12th International Conference on Computational Linguistics*, pp. 96-101.

Davison, Alice (2006), *Syntactic effects of correlative clause features in Sanskrit and Hindi/Urdu*, University of Iowa.

Delbruck, B. (1888), *Altindische Syntax*, Halle a.S, Germany: Verlag der Buchhandlung des Waisenhauses.

Evans, Nicholas (2008), "Reciprocal Constructions: Towards a structural typology", in Konig & Gast (eds.) *Reciprocals and Semantic Typology*, Amsterdam: Benjamins.

Faltz, Leonard (1985), *Reflexivity: A Study in Universtal Syntax*, New York: Garland.

Gillon, Brendan (1996), "Word order in Classical Sanskrit" *Indian Linguistics*: v. 57, n. 1, pp. 1-35.

Gillon, Brendan & Shaer, Benjamin, (2005), "Classical Sanskrit, `wild trees', and the properties of free word order languages" in Kiss, Katalin (ed.) *Classical languages and generative linguistics*, pp. 281—317, John Benjamins.

Gopal, Madhav, Mishra, D. & Singh, D. P. (2010), "Evaluating Tagsets for Sanskrit", in Jha, Girish Nath (ed.) *Proceedings of the Fourth International Sanskrit Computational Linguistics Symposium*, LNCS 6465, pp. 150–161, Heidelberg: Springer.

Goyal, Pawan, Vipul Arora & Laxmidhar Behera (2007), "Analysis of Sanskrit Text: Parsing and Semantic Relations", in Gerard Huet & Amba Kulkarni (eds.) *Proceedings of First International Symposium on Sanskrit Computational Linguistics*, pp. 23-36, INRIA, Paris.

Grishman, R. (1986) *Computational Linguistics: An Introduction*, Cambridge University Press: New York.

Gundel, J.K., N. Hedberg & R. Zacharski (1993), "Cognitive status and the form of referring expressions in discourse", in *Language*, 69(2):274-307.

Hellwig, O. (2007), "SanskritTagger, a stochastic lexical and POS tagger for Sanskrit", in G. Huet & A. Kulkarni (eds.) *Proceedings, First International Symposium on Sanskrit Computational Linguistics*, pages 37–46, INRIA, Paris.

Hobbs, Jerry (1978), "Resolving Pronoun References", *Lingua* 44, pp. 311-338.

Hock, H.H., ed. (1991) *Studies in Sanskrit syntax: a volume in honour of the centennial of Speijer's Sanskrit syntax*, Delhi: MotilalBanarsidass Publishers Pvt. Ltd.

Huet, G. (2006), "Lexicon-directed Segmentation and Tagging of Sanskrit", in Tikkanen, B., & Hettrich, H. (eds.) *Themes and Tasks in Old and Middle Indo-Aryan Linguistics*, Motilal Banarsidass, Delhi, pp. 307–325.

Huet, G. (2009), "Formal structure of Sanskrit text: Requirements analysis for a Mechanical Sanskrit Processor" in Huet, G., Kulkarni, A., & Scharf, P. (eds.) *Sanskrit Computational Linguistics* 2007/2008. LNCS (LNAI), vol. 5402, pp. 162–199. Heidelberg: Springer.

Jha, Girish N. et al. (2008), "Anaphors in Sanskrit", in Christer Johansson (ed.) *Proceedings of the Second Workshop on Anaphora Resoplution (WAR-II)*, NEALT Proceedings Series Vol, 2. pp. 11–25, Tartu University Library (Estonia): NEALT.

Jha, Girish N., Sobha, L., & D. Mishra (2009), "Discourse Anaphor and Resolution Techniques in Sanskrit", in Sobha L., A. Branco & R. Mitkov (eds.) *Proceedings of*

the 7th Discourse Anaphora and Anaphora Resolution Colloquium (DAARC 2009), pp. 135–150, Chennai: AU-KBC Research Centre.

Jha, Girish N., M. Gopal, & D. Mishra (2009), "Annotating Sanskrit Corpus: adapting IL-POSTS", in Z. Vetulani (ed.) *Proceedings of the 4th Language and Technology Conference: Human Language Technologies as a challenge for Computer Science and Linguistics*, November 6-8, Poznan, Poland: WydawnictwoPoznanskie Sp. Z o. o.

Johansson, Christer ed. (2007), *Proceedings from the first Bergen Workshop on Anaphora Resolution (WAR I)*, Cambridge Scholarly Publishing, UK.

Johnsbaugh, Richard & Marcus Schaefer (2004), *Algorithms,* Singapore-New Delhi: Pearson Education Pvt. Ltd.

Jurafsky, Daniel & James H. Martin (2000), *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*, NJ: Englewood Cliffs, Prentice Hall.

Kale, M.R. (1961), *A Higher Sanskrit Grammar*, New Delhi: MLBD Publishers.

Keith, A.B. (1993), *A History of Sanskrit Literature*, New Delhi: MLBD Publishers.

Kennedy, Christopher & Branimir Boguraev (1996), "Anaphora for Everyone: Pronominal Anaphora Resolution without a Parser", in *Proceedings of the 16th International Conference on Computational Linguistics (COLING'96)*, pp. 113-118, Denmark.

Kentor, R.N. (1977), *The Management and Comprehension of Discourse Connection by Pronouns in English*, Ph. D. thesis, Department of Linguistics, Ohio State University.

Khanna, Harish (1996), "Is Hindi a configurational language", in Lakshmi, V.S. & Aditi Mukherjee (eds.) *Word Order in Indian Languages*, pp. 115-139, Hyderabad: Booklinks Corporation

Kumar, Ela (2003), *Design and Development of Pronominal Anaphora Resolution System*, unpub. Ph.D. thesis, Delhi: University of Delhi.

Koster, Jan & Eric Reuland, eds. (1991), *Long-distance anaphora*, Cambridge: Cambridge University Press.

Lappin, Shalom & Herbert J. Leass (1994), "An Algorithm for Pronominal Anaphora Resolution", in *Computational Linguistics*, 20, 4, 535-561.

Lasnik, Howard ed. (1989), *Essays on Anaphora*, Dordrecht: Kluwer.

Liang, Y.D. (2007), *Introduction to Java Programming*, NJ: Pearson Education, Inc.

Lust, Barbara C., Kashi Wali, James W. Gair, & K.V. Subbarao, eds. (2000), Lexical anaphors and pronouns in selected south Asian languages: a principled typology, New York: Mouton de Gruyter.

McCord, Michael C. (1990), "Slot Grammar: A System for Simpler Construction of Practical Natural Grammars", in *Proceedings of the* International Symposium on Natural Language and Logic, pp. 118-145. London: Springer-Verlag.

Mitkov, Ruslan (1998), "Robust Pronoun Resolution with Limited Knowledge", in Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, ACL-36, pp. 869-875. Association for Computational Lingistics, Morristown, NJ, USA.

Mitkov, Ruslan ed. (2003), *The Oxford Handbook of Computational Linguistics*, New York: Oxford University Press.

Mitkov, Ruslan, Boguraev Branimir, & Lappin Shalom (2001), "Introduction to the special issue on Coputational Anaphora Resolution", in proceedings of *ACL 2001*.

Mitkov, Ruslan (1997), "Factors in Anaphora Resolution: They are not the only Things That Matter. A Case Study Based on Two Different Approaches", in *Proceedings of the ACL'97/EACL'97 Workshop on Operational Factors in Practical, RobuAnaphora Resolution*, 14-21, Spain.

Mitkov, Ruslan, Lamia Belguith & Malgorzata Stys (1998), "Multilingual RobuAnaphora Resolution", in *Proceedings of the Third International Conference on Empirical Methods in Natural Language Processing (EMNLP-3)*, 7-16, Spain.

Murthy, K.N., Sobha L., & Muthukumari B. (2005), "Pronominal Resolution in Tamil Using Machine Learning Approach", in Christer Johansson (ed.) *Anaphora Resolution Using Machine Learning*.

Pralayankar, P. & Sobha L. (2010), "Anaphora Resolution Algorithm for Sanskrit", in Jha, Girish Nath (ed.) *Proceedings of the Fourth International Sanskrit Computational Linguistics Symposium*, LNCS 6465, pp. 209–217, Heidelberg: Springer.

Prasad, Rashmi & Michael Strube (2000), "Discourse Salience and Pronoun Resolution in Hindi", in Penn Working Papers in Linguistics 6.3, pp. 189-208.

Schwarz-Friesel, Monika, Consten, Manfred, and Knees, Mareille (2007), *Anaphors in text: cognitive, formal and applied approaches to anaphoric reference.* Amsterdam: Benjamins.

Sobha, L. & B. N. Patnaik (1998), "An Algorithm for Pronoun and Reflexive Resolution in Malayalam", in *Proceedings of the International Conference on Computational Linguistics, Speech and Document processing*, pp. 63-66.

Sobha, L. (2008), "Anaphora Resolution Using Named Entity and Ontology", in Christer Johansson (ed.) *Proceedings of the Second Workshop on Anaphora Resoplution (WAR-II)*, NEALT Proceedings Series Vol, 2. Tartu University Library (Estonia): NEALT.

Speijer, J.S. (1886, repr. 2006), *Sanskrit Syntax*, New Delhi: MotilalBanarsidass Pvt. Ltd.

Staal, J.F. (1967), *Word Order in Sanskrit and Universal Grammar,*Holland: D. Reidel Publishing Co.

Subbarao, K.V. (ms.) *South Asian Languages: A Synactic Typology.*

Upadhyay, B. (2001), *Sanskrit Sahitya ka Itihas*, Varanasi: Sharda Niketan.

Uppalupu, Bhargav & Dipti Mishra Sharma (2009), "Pronoun Resolution in Hindi", in Sobha L., A. Branco and R. Mitkov (eds.) *Proceedings of the 7th Discourse Anaphora and Anaphora Resolution Colloquium (DAARC 2009),*Chennai: AU-KBC Research Centre.

Whitney, W.D. (1889, repr. 2004), *Sanskrit Grammar*, New Delhi: Munshiram Manoharlal Publishers Pvt. Ltd.

Web References

AU-KBC Research Centre - http://www.au-kbc.org/frameresearch.html (accessed: 25 April 2011).

Computational Linguistic R&D, J.N.U., *http://sanskrit.jnu.ac.in/index.jsp* (accessed: 18 May 2011).

Characteristics   of   Sanskrit, <http://murthygss.tripod.com/some_characteristics_of_sanskrit.htm> (accessed: 5 July 2011).

Department of Sanskrit UOHYD >> sandhi
http://sanskrit.uohyd.ernet.in/~anusaaraka/sanskrit/samsaadhanii/sandhi/index.html
(accessed: 5 April, 2011).

Java, Servlet, http://java.sun.com/products/servlet/ (accessed: 5 June, 2011).

Java Server Pages, http://java.sun.com/products/jsp/ (accessed: 3 July, 2011).

John R. Pierce et al., *Language and Machines — Computers in Translation and Linguistics*. ALPAC report, National Academy of Sciences, National Research Council, Washington, DC, 1966. <http://www.nap.edu/books/ARC000005/html> (accessed: 29 June 2011).

Punctuation, <http://en.wikipedia.org/wiki/Punctuation> (accessed: 27 May 2011).

Saṃbhāṣaṇasaṃskṛtam Śabdakośaḥ, <http://spokensanskrit.de/> (accessed: 3 July 2011).
The Leipzig Glossing Rules, <http://grammar.ucsd.edu/courses/lign120/leipziggloss.pdf> (accessed: 5 March 2011).

The Web server, Apache Tomcat, http://www.apache.org/ (accessed: 1 July 2011).