Pronominal Anaphora Resolution in Bengali

Apurbalal Senapati

Pronominal Anaphora Resolution in Bengali

Thesis submitted in the partial fulfilment of
the requirement for the degree of

Doctor of Philosophy

in

Computer Science

by

Apurbalal Senapati

Advisor

Prof. Utpal Garain

Computer Vision and Pattern Recognition Unit
Indian Statistical Institute
203 B.T. Road, Kolkata - 700 108, India
May, 2016

CERTIFICATE

−−/−−/− − −−

This is to certify that the thesis entitled **Pronominal Anaphora Resolution in Bengali**, submitted by **Apurbalal Senapati**, Indian Statistical Institute, is a record of bona fide research work under my supervision and I consider it worthy of consideration for the award of the degree of Doctor of Philosophy of the Institute. This work has not been submitted earlier to any other Institute or University for any degree or diploma.

Prof. Utpal Garain
Professor
CVPR Unit
Indian Statistical Institute
203 B. T. Road
Kolkata - 700 108, India

DECLARATION

In accordance with the appropriate regulations, I, Mr. Apurbalal Senapati declare that:

a. The work contained in the thesis is original and has been done by myself under the general supervision of my supervisor.

b. The work has not been submitted to any other Institute for any degree or diploma.

c. I have followed the guidelines provided by the Institute in writing the thesis.

d. I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.

e. Whenever I have used materials (data, theoretical analysis, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references.

f. Whenever I have quoted written materials from other sources, I have put them under quotation marks and given due credit to the sources by citing them and giving required details in the references.

Apurbalal Senapati
Research Scholar
CVPR Unit
Indian Statistical Institute
203 B. T. Road
Kolkata - 700 108

Dedicated to the people who have sacrificed their lives to establish the truth.

ACKNOWLEDGEMENT

Several people have immensely helped me in completing this thesis and it is indeed difficult for me to enlist their names and contribution in a single paragraph. At first, I would like to thank my teacher and sole guide Prof. Utpal Garain for his dedication, encouragement and inspiration. After finishing my master's, I joined the software industry for a couple of years. Due to this break in my regular educational attachment, initially it was bit difficult for me to switch back into research, but it was possible only because of Prof. Garain's constant encouragement and help. Next, I reserve my special thanks for my wife Krishna Mallik and other family members for their mental support and releasing me from other family related issues as much as possible.

I am greatly thankful to the previous and the present heads of the CVPR Unit, Prof. Bidyut Baran Chaudhuri and Prof. Umapada Pal for their concern, encouragement, advice and support in all aspects. I feel very grateful to our Dean of Studies and the Institute Director for their kind help. I would like to express my deep gratitude to all the faculty members of CVPR unit for their help and for providing a healthy environment of research. I feel myself lucky enough to be accompanied in attending a course on Research Methodology provided by Prof. Palash Sarkar from Applied Statistics Unit, which was very helpful for every new comer in this area. I also thank to all the faculty members of RFAC (Research Fellows Advisory Committee) for their valuable suggestions on my work at the end of every year. It would be injustice if I don't mention the role of Mr. Shankar Sen and all the other office staff for providing me required logistics for conducting my research. I would like to pay special thanks to my colleagues Ankan, my juniors Arjun, Abhisek, Anabik and other lab members for their continuous support. I would like to express my special appreciation and thanks to my friend Baidya, and senior Dr. Raghunath Chatterjee, Mr. Subrata Halder for their valuable advice and support in critical situations.

I also express my deep respect to Prof. Soumitra Bondyapadhay, IISER Kolkata for his dedication to social activities, especially to build the scientific mind, objective thinking under the social organization, *Breakthrough Science Society*, which has always been a constant source of inspiration in my life. At last, I would like to express my gratitude to my well-wisher Sabita-*di* for her continuous inspiration and support.

Apurbalal Senapati

Contents

List of Figures

List of Tables

List of Symbols and Abbreviations

List of Abbreviations

AR	Anaphora Resolution
ABP	Anandabazar Patrika
CC	Conjuncts
CL	Computational Linguistics
EM	Expectation Maximization
ENGCG	English Constraint Grammar
FIRE	Forum for Information Retrieval Evaluation
IR	Information Retrieval
LL	Log Likelihood
LR	Language Resource
MARS	Mitkov's Anaphora Resolution System
MLP	Multi-Layer Perceptrons
MUC	Message Understanding Conference
NER	Named Entity Recognition
NLP	Natural Language Process
NN	Noun

NNP	Proper Noun
NP	Noun Phrase
PAR	Pronominal Anaphora Resolution
PEA	Pronoun Emitting Approach
POS	Parts of Speech
PP	Prepositional Phrase
PRP	Pronoun
QA	Question Answering
TDIL	Technology Development for Indian Languages

Chapter 1

Introduction

1.1 Introduction

In linguistics, the term **anaphora** is used to refer to a relation between two linguistic elements and the interpretation of one depends on the interpretation of preceding linguistic context [76]. Actually, the term **anaphora** is derived from an ancient Greek word ($\alpha\nu\alpha\phi o\rho\alpha$) which means "*the act of carrying back upstream*". In computational linguistics (CL), the term **anaphor** is a reference which points back to a word or phrase that has been mentioned earlier in the text being processed. The referred word or phrase is known as the **antecedent**. The process of identifying anaphoric relations is known as **anaphora resolution**. In contrast, the **cataphora** points to the expression in forward direction. Generally, the anaphora and cataphora are known as an **endophora** and where, **exophora** refers to something outside of the text. Consider the following examples:

- S1: The *students* studied hard for *their* final examination.

- S2: Because *she* studied really hard, *Smrity* got the first position in the test.

- S3: Arjun bought a $laptop_i$ online, but it_i was defective, please return it to *them* immediately.

In sentence S1, *their* is an anaphora, and it refers to *students* in backward direction and hence *students* is the antecedent. Where in the sentence S2, *she* is a cataphora and refers to *Smrity* in the forward direction. In sentence S3, the referent of *it* is *laptop* but the referent of *them* should be the online vendor, but not present in the text and hence is an exophora.

The anaphora has been studied for decades under various disciplines ([77], [78], [79], [80], [81], [82]). The linguistic people ([83], [84], [85], [86], [87], [88]) have investigated it within the syntactic and semantic constraints of language whereas the computational linguists use the linguistic knowledge to study the computational aspects i.e. how anaphora can be resolved by computers. The Anaphora Resolution (AR) plays an important role in several Natural Language Processing (NLP) applications including information extraction, question answering, text summarization, etc. to detect the relations among entities. Sometimes the meaning of a sentence, in the text is not clear without resolving the anaphora and hence AR is a general problem in NLP domain. This thesis is related to AR in Bengali. It deals with the major computational issues in AR. The thesis adopted an existing state-of-the-art system GuiTAR (which was originally developed for English) for Bengali and developed a new algorithm named as “Pronoun Emitting Approach” to resolve the anaphora. Finally, the thesis outlined some advanced issues related to anaphora resolution.

1.2 Background

The research on Anaphora Resolution is more than thirty-five years old in the field of Computational Linguistics. But the problem is still considered as a challenging problem in NLP and related areas like Question Answering (QA), Machine Translation (MT), etc. The research in AR has been done extensively in English, with several promising approaches and different implementations of the task [89]; some approaches exist for other European languages, like Dutch, German, Italian, and Spanish (SemEval-2010 [90]) etc. In many cases, researchers tried to develop AR systems simply by following the existing approaches. Most of the algorithms can be classified as one of the two main types, namely rule-based and statistical. The first category comprises those algorithms that use extensive domain and linguistic knowledge (Brennan et. al. [83], Strube [91], Tetreaul [92]) or use a salience-based strategy based on syntax with limited usage of linguistic and domain knowledge (Kennedy and Boguraev [93], Baldwin [16], Mitkov [94], Palomar et. al. [95], Mitkov [8], Trouilleux [96]). In case of machine learning based approaches [97] a large set of tagged data is required to train the system.

In the recent years, the main focus of research in this domain in English has exhibited a shift from anaphora resolution to co-reference resolution. Similarly, in order to get rid of language dependency it has shifted from heuristic approaches to statistical ones [98]. And for the resource scarce languages (like Indian or many other Asian languages) AR is still an active research problem.

There are several languages in India belongs to different language families and the research in AR in Indian languages is very limited compared to English and other European languages. A few years back, ICON-2011 [14] paid a special attention to AR in Indian languages. The NLP tool contest on anaphora resolution was attempted for three Indic languages (Hindi, Bangla and Tamil) and annotated data was provided. Indian languages suffer from lack of required resources like Part of Speech (POS) tagger, chunker, parser etc. and annotated corpus. Though people are trying to overcome the limitations by designing unsupervised learning methods, but unavailability of properly annotated data is still a major stumbling block to do this.

1.3 Problem statement

Anaphora can be divided into different types according to the syntax (i.e., based on their form) or semantic (based on the anaphor-antecedent relationship). But all types are not frequently used in the language. Most widespread type is pronominal (anaphora which is realized by anaphoric pronouns [8]) is in the CL literature; and hence in this thesis, we mainly concentrate on Pronominal Anaphora Resolution (PAR). Most of the Indian Languages are inflectional in nature and Bengali is one such highly inflectional language and hence Bengali has a rich inflectional morphology. And for the inflectional variation there is a large set of pronouns in Bengali. In Pronominal Anaphora Resolution the study of pronoun is obviously a sub-task in Bengali. This study shows that pronouns exhibit very complex characteristics in Bengali and therefore, it's a challenging research problem. However, very little research has been done to address these challenges.

Our work is dedicated to (i) identify the computational issues of pronominal AR for Bengali and corpus based study of Bengali pronouns (ii) provide the basic NLP tools for AR and tagged data for AR (iii) identify the dependencies to adopt an existing English-based system (iv) build a new framework outperforming the other existing approaches and finally (v) enlighten on future research issues.

1.4 State of the art

The research in anaphora resolution evolves from simple to complex in the context of resources and algorithms used. Most of the works in early seventy of the previous century were based on heuristic rules along with some linguistic knowledge. Some of the such important studies are reported by Bobrow [99], Winograd [100], Woods [101], Hobbs ([102], [103]), etc. Later on, in many NLP tasks the study of anaphora resolution was more theoretically oriented and researchers tried to incorporate more features. Mitkov ([89], [8]) explained these efforts in details and the survey paper by Massimo et. al. [104]

has addressed approaches to anaphora covering the linguistic and psycholinguistic issues, machine learning methods and lexical and encyclopaedic knowledge. The most prominent research efforts are briefly outlined next. For each method a summary of description of the features, the resolution methods and evaluation techniques are highlighted.

1.4.1 Previous works

Bobrow (1964): A heuristic approach was followed in the earliest system STUDENT [99] to resolve anaphora. The system was dedicated to solving the simple mathematical problems like high school algebra and *wh-* questions. The system used pattern matching approach (matching of a phrase) for finding antecedents of anaphora. Later on, Hirst [105] pointed out the limitation of the Bobrow's system with examples. For instance, if the same problem describes with different sentence structure, then the system might fail.

Winograd (1972): The system SHRDLU [100], a pronoun resolution system used heuristic rules like STUDENT. This system handled much more complex anaphoric expressions as compared to STUDENT and addressed dialogues and zero anaphora too. The system considered not only the previous noun phrases or first likely candidate, but considered other possible antecedents. The system considered syntactic position, i.e. the subject is favoured over the object. The concept of "focus" element were incorporated and "focus" elements were favoured. The focus being identified from the answers to *wh-* questions.

Woods (1972): The system LUNAR [101], originally developed to support NASA for analysis of the lunar geological data for rock and soil. Later on it turned as a natural language understanding system, specially for the geologist asking questions regarding chemical testing such as "What is the average concentration of Aluminium in high-alkali rocks?". It used the Augmented Transition Network (ATN) grammar and semantic information. The system handled two classes of anaphora namely, partial and complete. The complete anaphora implied the complete NPs as antecedent, while the partial anaphora referred to the parts of preceding NPs. Unlike STUDENT system, this approach operated at syntactic and semantic levels rather than at the lower level of lexical matching. Though the whole system was never fully tested, but an evaluation of a preliminary version was reported in [106]. The 78% of the questions asked to the system were understood and responded correctly, 12% failed due to trivial clerical errors like dictionary coding errors, etc. The rest 10% of the questions failed because of semantic and parsing error.

Hobbs (1976, 1978): One of the earliest algorithms developed by Hobbs ([102], [103]) is based on various syntactic knowledge. The algorithm uses a surface parsed tree structure identifying the subject, verb, object, etc. and uses simple syntactic agreement and properties of pronoun to select its antecedent. The algorithm traverses the surface parse tree left-to-right, breadth-first manner and then goes backwards one sentence at a time, looking for an antecedent matching the pronoun in gender and number. Several assumptions have been made about the sentence structure. It assumes that the parse tree represents the correct grammatical structure of the sentence and also assumes that the NP node has an $\bar{N}$ node below it, with $\bar{N}$ denotes a noun phrase without its determiner. Based on these assumptions, Hobbs explained the distinction between the following two sentences ([102], [103]):

- S4: Mr. Smith saw a *driver* in *his* truck.
- S5: Mr. Smith saw a driver of his truck.

The structures of the sentences assumed for the relevant noun phrases are in shown in Figure 1.1 (a) and (b) respectively. In (a) *his* may refer to the *driver*, but it may not in (b).

The system was evaluated with 300 pronouns (*he* = 139, *she* = 7, *it* = 71, *they* = 83) from three different texts consisting of simple to complex sentences in terms of

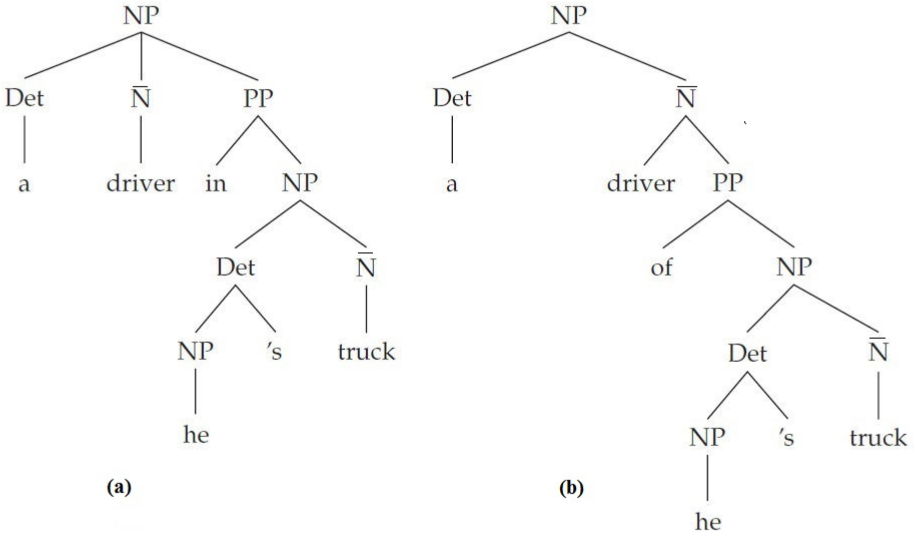


Figure 1.1: Parse tree corresponding to relevant noun phrases of (S4) and (S5)

grammatical features. Two heuristic rules were used for finding the antecedent. The first is the hypothesis that the antecedent is always found within the last $\boldsymbol{n}$ sentences for some small values of $\boldsymbol{n}$ ([107], [108]), but the problem was to find out the suitable value of $\boldsymbol{n}$. It was observed that the majority of the antecedents is found with $\boldsymbol{n} = \boldsymbol{0}$ (271 out of 300). The second heuristic is, "if the same pronoun occurs twice in the same sentence or in two consecutive sentences, the occurrences are coreferential" [100]. But the performance of the heuristic was not impressive, it returned correct antecedent only 28 times out of 48 cases. The overall performance of the algorithm was 88.33% (265 out of 300) accuracy. The result also shows that with the addition selectional restrictions it achieved 91.66% (275 out of 300) accuracy. The success rate has been somewhat deceptive since in over half of the cases there was only one plausible antecedent. And hence a separate analysis of the algorithm was done in the case of conflict (more than one plausible antecedent occurred in the candidate set). Of 132 such cases, 12 were resolved by selectional restrictions, and 96 of the remaining 120 were resolved by the algorithm. Thus, 81.82% (108 out of 132) of these 'conflicts' were resolved by a combination of the algorithm and the selectional restrictions. This algorithm is still often used as a baseline to compare with another algorithm.

Sidner (1979): The key concept of the Sidner's work [88] is to find the local focus. So changes in focus through a discourse was modelled to identify antecedents. Two terms namely, *discourse focus* (DF) and *actor focus* (AF) were defined. The assumption on focus is that, at a given point, a well-formed discourse is about some entity mentioned; this entity is called the discourse focus. "An actor is an animate object which can function as the agent of a particular verb. Actors can be the focus of the discourse". The *discourse focus stack* (DFS) and *actor focus stack* (AFS) are registers each contain a list of zero or more entities. The Sidner's algorithm can be described by the steps: (a) identify the focus, i.e. proposes a technique for tracking the discourse focus; (b) design a method for assigning antecedents of definite pronouns for the focus and (c) update the focus registers, considering the results of anaphora interpretation already done. The algorithm makes an initial prediction of the focus after scanning the first sentence and using syntactic and semantic criteria. In other sentences, an anaphora interpretation algorithm is used to resolve each anaphora. There is a set of rules to suggest one or more antecedents based on the belonging of the focus registers. Once the focus registers have been updated, then the anaphora interpretation algorithm is used for the next sentence. Sidner explained the algorithm using several examples instead of presenting any formal evaluation, but the concept had the greatest influence on subsequent research in the Centering theory.

Carter (1986, 1987): The Carter's works ([109], [110]) are only restricted to nominal anaphora (referring expression has a non-pronominal noun phrase as its antecedent) and a " shallow processing" approach has been used to resolve anaphora. He pointed out that particularly in anaphora resolution, the use of linguistic knowledge is relatively cheaper than the use of the domain and common-sense knowledge in terms of computational complexity. And hence he tries to restrict the use of detailed domain and common-sense knowledge and inference by exploiting general linguistic knowledge as much as possible. If the anaphora cannot be resolved by shallow processing completely, then domain knowledge is sparingly used. The main attractive aspect of the work is that it builds on existing work where appropriate, and extends it where required. For example, it integrates the parsing from Boguraevs [111] work, the theory of local focusing from Sidner [88] work and the preference semantics and common-sense inference have been used from Wilks [184]'s work. Carter's system is implemented in SPAR (Shallow Processing Anaphora Resolver) as part of his PhD thesis, and the system SPAR was evaluated on simple 60 small English stories covering a variety of topics. The accuracy of the system is 93% for pronominal anaphora (226 out of 242) and 82% for non-pronominal anaphora (65 out of 80). This result was the highest success rate obtained by anaphora resolution programs at that time.

Rich and LuperFoy(1988): The study in [113] describes the pronominal anaphora resolution module of Lucy. Lucy is a portable English understanding system consisting of the components like a syntax-based parser [114], a semantic translation system, a pronominal anaphora resolution system and a pragmatic processor. The work is concentrated on the pronominal anaphora resolution module. The observation was that, though there were many theories for anaphora resolution exist, but no one of these theories is complete. Each partial theory accounts for a subset of the facts that influence the process of anaphora resolution. This scenario motivated them to implement a blackboard-like distributed architecture [115] in which individual partial theories can be encoded as separate modules that can interact to propose candidate antecedents and to evaluate each other's proposals. The distributed architecture consists of a set of loosely coupled modules, where each module handles a subset of discourse phenomena by implementing a specific partial theory. These modules communicate by proposing candidate antecedents and by evaluating each other's proposals. The handler, a specially designated oversight module, mediates these communications and resolve conflicts among the modules (Figure 1.2). The ovals in the figure represent the implementation of one of the partial theories of anaphora and each of these implementations is called a constraint source.

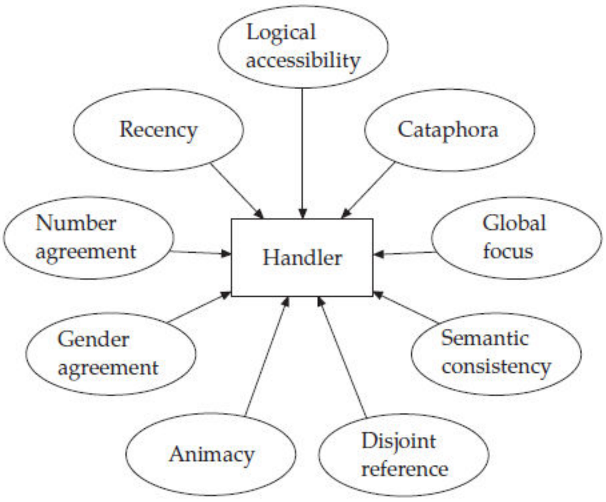


Figure 1.2: Rich and LuperFoy's Architecture [8]

The important contribution of the work is the analysis of how factors can interact and influence the decision on the antecedent. The selection of the antecedent from a set of candidates is made on the basis of a combined score resulting from the examination of each candidate by the entire set of factors. The score is a number in the range -5 to +5, the confidence is a number in the range 0 to 1. The formula that combines a set of $\boldsymbol{n}$ (score, confidence) pairs is:

$$\text{Running score} = \frac{\sum_{i=1}^{n} score(i) \times confedence(i)}{\sum_{i=1}^{n} confedence(i)} \tag{1.1}$$

This equation computes an average which is weighted not by the number of distinct scores, but by the total confidence expressed for the scores. The factors that wish to offer no opinion can simply suggest a confidence of 0 to its opinion, which, in turn, will have no effect on the running score of a candidate. The factors are implemented in one of the following four categories: Finite set generators (such as disjoint reference when applied to a reflexive pronoun), Fading infinite set generators (such as recency), Filters (such as number and gender agreement) and Preferences (such as semantic content consistency). The major shortcoming of this study is that, it did not report any evaluation result.

Carbonell and Brown (1988): Philosophically this approach [116] is almost similar to Rich and LuperFoy [113] in combining multiple strategies. The study also focuses on the problem of inter-sentential anaphora [117]. The evidence shows that inter-sentential anaphora is more frequent and crucial in language processing [118]. The system applies the knowledge sources like constraints and preferences. The first set of constraints is applied to filter the candidate list and then preferences are applied to the refined candidates. The knowledge sources are used like sentential syntax, case-frame semantics, dialogue structure and general world knowledge, etc. The constraints are generally considered as the local syntax agreements (number, gender, case, etc.), case-role semantic constraints (semantic features), pre-condition/post-condition constraints (real-world and pragmatics knowledge), case role persistence preference, semantic alignment preference, syntactic parallelism preference, syntactic topicalisation preference (favours topicalised candidates), inter-sentential recency preference. In actual implementation the framework uses a parsed input [119] with syntactic as well as semantic knowledge and a modified form of lexical-functional grammar [120] unifying semantics and semantic knowledge sources to generate a complete parse of each sentence. The algorithm first selects the best anaphoric referent among several candidates and then applies the resolution strategy. The possible candidates are considered from the previous sentence and if no suitable candidates found in the previous sentence, then consider the one before and so on. But the number of sentences examined may be changed depending on other discourse phenomenon. In case of conflict (more than one candidate passes through the constraints and preferences) gives the ambiguous solution. Also, in addition to eliminate the candidates, semantic and local anaphora constraints are participating to cast votes for eligible candidates matching to the anaphora. In absence of hard constraints, preferences are applied. The preferences use a voting score to determine which candidate referent is most preferred. Individual weight is assigned for each preference strategy, and may vote with less than its full weight for less preferred candidates. The evaluation was reported on a small sample data set. The sample consists of 31 sentences containing 27 pronouns and 4 lexical NP anaphora. The anaphora resolver shows 78% success rate.

Dagan and Itai (1990, 1991): The approach ([121], [122]) is based on an automatic scheme for collecting statistics on co-occurrence patterns in a large corpus. The main concern of the work is to suggest an alternative to the traditional model, based on the automatic acquisition of constraints from a large corpus, and to show how this method is used to resolve anaphora references. Selectional constraints are used in anaphora resolution i.e. the antecedent must satisfy the constraints imposed on the anaphora. In this model, each of the candidates is substituted with the anaphora and only those candidates which produce frequent co-occurrence patterns are considered and the candidate that produces the most frequent co-occurrence patterns is preferred. The author illustrated the concept with an example taken from the Hansard corpus [116] of the proceedings of the Canadian parliament.

- "They know full well that the companies held tax money aside for collection later on the basis that the government said *it* was going to collect *it*".

There are two occurrences of *it* in the above sentence. The first is the subject of collect and the second is its object. Statistics are gathered for the three possible candidates for antecedents in this sentence: *money*, *collection* and *government.* Table 1.1 shows the patterns produced by replacing each candidate with the anaphora, and the number of times each of these patterns occurred in the corpus. According to these statistics, *government* is preferred as the antecedent of the first *it* (which is in subject position in the sentence), and *money* of the second *it* (which is in the object position in the sentence).

The use of the statistical model involves two separate phases; the *acquisition phrase* and the *disambiguation phase.* In the acquisition phase, the corpus is processed and the

Table 1.1: Statistics of co-occurrence patterns associated with the verb *collect* based on an excerpt from the Hansard corpus

subject–verb	collection	collect	0
subject–verb	money	collect	5
subject–verb	government	collect	198
verb–object	collect	collection	0
verb–object	collect	money	149
verb–object	collect	government	0

statistical database is built and in the disambiguation phase, the statistical database is used to resolve ambiguities. The statistical database contains co-occurrence patterns for the following three pairs of syntactic relations: *subject–verb*, *verb–object* and *adjective–noun*. To identify these relations, sentences are parsed by the PEG parser [123]. An experiment was performed to resolve anaphoric *it* in the Hansard corpus [116]. The test data was manually selected from the corpus and the experiment was conducted on 59 examples. The statistics were collected from a part of the corpus consisting of 28 million words. For 21 out of the 59 examples the statistics were not meaningful because the threshold of 5 occurrences per alternative could not be reached. In the remaining 38 examples, the method found the correct antecedent 33 times (87% of the cases).

Lappin and Leass (1994): Lappin and Leass presented an algorithm for resolving pronouns is known as Resolution of Anaphora Procedure (RAP) [124]. The algorithm is based on syntactic structure of McCord's Slot Grammar [86] and the salience scores derived from syntactic structure and corresponding to a set of factors. The preference for a candidate is considered via salience scores, it does not employ the semantic information or real-world knowledge to select the candidates from the candidate list. It does not incorporate the semantic information or real-world knowledge in choosing from the candidates. The salience scores are calculated corresponding to a set of anaphora factors. In the algorithm the following factors has been considered:

The Syntactic Filter Pronoun-Noun Coreference : A pronoun (PRP) is not co-referential with a (non-reflexive or non-reciprocal) noun (N) if any of the six conditions hold: (i) PRP and N have incompatible agreement features; (ii) PRP is in the argument domain of N; (iii) PRP is in the adjunct domain of NP; (iv) PRP is an argument of a head H, NP is not a pronoun, and NP is contained in H; (v) PRP is in the noun phrase domain of N; (vi) P is a determiner of a noun Q, and NP is contained in Q.

Test for Pleonastic Pronouns: syntactic and semantic tests are used to determine the pleonastic "it". A class of modal adjectives (such as necessary, possible, certain, likely, difficult, legal, etc.) and a class of cognitive verbs (such as recommend, think, believe, etc.) to identify the pleonastic it.

Anaphora Binding Algorithm: it uses the following hierarchy of 'argument slots' subject > agent > object > (i-object - p-object). Where the subject is the surface subject as identified by the slot grammar parser, agent is the deep subject of a verb heading a passive VP, object is the direct object slot, i-object is the indirect object, and p-object is the object of a PP complement of a verb.

Salience Weights: salience weights are computed on the basis of salience factors like grammatical role, parallelism of grammatical roles, frequency, proximity and recency. The salience factors and their weights are given in Table 1.2.

Equivalence Classes: identifying the co-referential (anaphoric) chain i.e. the linked NPs for which a global salience score is computed as the sum of the salience value of its constituents.

Table 1.2: Salience factor types with initial weights

Factor type	Initial weight
Sentence recency	100
Subject emphasis	80
Existential emphasis	70
Accusative emphasis	50
Indirect object and oblique complement emphasis	40
Head noun emphasis	80
Non-adverbial emphasis	50

The system was tested on a corpus of five computer manuals containing approximately 82,000 words. The corpus had a total of 560 third person pronouns, including reflexives and reciprocals. In the training phase, they experimented a lot with salience weighting in order to optimize RAP's success rate. They heuristically analysed the failure cases to eliminate them in a general manner. Here they rewarded the parallelism (the antecedent with respect to the same function that the anaphor is selected) and it substantially improves the results. The salience factor that was originally present, viz. matrix emphasis, was revised and known as non-adverbial emphasis factor. Here the number of cases that the RAP resolves correctly was 475 i.e. success rate was 85%. A blind test also was performed on 360 pronoun occurrences, of a corpus randomly selected from computer manuals containing 1.25 million words. The algorithm performed successful resolution at 86% of the cases.

Kennedy and Boguraev (1996): The algorithm is described by Kennedy and Boguraev [93] is the modified and extended version of the method developed by Lappin and Leass [124]. The algorithm does not require in-depth, full, syntactic parsing of text; instead they used a parts of speech tagger, enriched only with annotations of grammatical function of lexical items in the input text stream. The reason for not using the parser is to reduce pre-processing errors. They used the output of high yielding accuracy of ENGCG POS tagger ([125], [126]), augmented with syntactic function annotations for each input token. The morpho-syntactic tagging system ([125], [126]) does such types of analysis. The tagger provides a very simple analysis of the structure of the sentences in the text. For each lexical item in each sentence, it augments a set of values which indicate the morphological, lexical, grammatical and syntactic features of the item in the context in which it appears.

The basic logic of the algorithm is almost similar to that of the Lappin and Leass algorithm [124]. It scans the text in sentence by sentence from left to right fashion. The discourse referents are interpreted as: either it is taken to introduce a new participant in the discourse or it is taken to refer to an earlier interpreted discourse referent. The resolution is done by filtering the anaphoric expression that cannot be possibly refer, then selecting the optimal antecedent from the remaining candidates, where optimality is determined by a salience measure. For implementation, they have considered "COREF classes" which is the equivalence classes of anaphorically related discourse referents. The COREF class is represented as an object which contains all necessary information about the COREF class as a whole, including, membership, salience. The relationship between a discourse referent and its COREF class is mediated through the COREF object as follow. For every discourse referent includes an informative parameter which is a pointer to a COREF object; discourse referents which have been considered to be co-referential by sharing the same COREF value. The salience factors used by Kennedy and Boguraev are almost similar to the Lappin and Leass. In addition, three more factors are included (Table 1.3). The salience factors possessive (POSS-S), which rewards discourse referents whose grammatical function (GFUN) is possessive, and context (CNTX-S), which boosts the score of candidates that appear in the same discourse segment as the anaphora. The individual salience factors are associated with numerical values; and the overall salience of a COREF is the aggregate of the values of the salience factors that are satisfied by some member (satisfied at most once by each member of the class) of the COREF. The salience factors used by the algorithm are given in Table 1.3.

Table 1.3: Salience factors used by Kennedy and Boguraev

Factor	Initial weight
SENT-S (sentence recency):	100 iff in the current sentence
CNTX-S (context emphasis):	50 iff in the current context
SUBJ-S (subject emphasis):	80 iff GFUN = subject
EXST-S (existential emphasis) :	70 iff in an existential construction
POSS-S (possessive emphasis):	65 iff GFUN = possessive
ACC-S (accusative emphasis) :	50 iff GFUN = direct object
DAT-S (indirect object emphasis) :	40 iff GFUN = indirect object
OBLQ-S (oblique complement emphasis) :	30 iff the complement of a preposition
HEAD-S (head noun emphasis) :	80 iff EMBED = NIL
ARG-S (non-adverbial emphasis) :	50 iff ADJUNCT = NIL

There is a significant difference with the approach of Lappin and Leass [124] in the determination of disjoint reference. Since Kennedy and Boguraev did not use the syntactic parse information, instead they rely on the configuration relations which play a prominent role in determining which constituents in a sentence a pronoun may refer to. The following three configurational constraints have been exploited in the algorithm.
Condition 1: A pronoun cannot co-refer with a co-argument.
Condition 2: A pronoun cannot co-refer with a non-pronominal constituent when it both commands and precedes.
Condition 3: A pronoun cannot co-refer with a constituent which contains it.
The evaluation was done on a dataset of 27 texts taken from a random selection of genres including newspaper articles, magazine articles, product announcements, and news stories World Wide Web pages. The data consisted of 306 third person pronouns out of which 231 (75%) were correctly resolved. Note that the result (86% accuracy) obtained from the data used by Lappin and Leys [124] is a single text genre only. The experimental result shows that it is possible to attain comparable levels of accuracy (as compared to RAP) without the use of a complex parser.

Baldwin (1997): Baldwin developed a rule based high precision pronoun resolution system named CogNIAC [16]. The system was designed based on the assumption that there is a sub-set of pronominal anaphora that does not require common-sense knowledge or general purpose reasoning. The system requires several pre-processing including sentence detection, parts-of-speech tagging, simple noun phrase recognition, basic semantic category information like, gender, number, and in one configuration, partial parse trees. The system uses only eight (six high precision and two low precision) domain independent, high confidence, simple and ordered rules. The algorithm does not resolve the pronouns in case of ambiguity and left out unresolved when it does not follow any one of the rules in the rule base. Hence, the result produced by the system is very high precision, but unsatisfactory recall. The rules are ordered implies that, the rules are applied one by one in sequential order. Once an antecedent is found for a specific rule, then no further rules are applied and if no rules resolve the pronoun, and then it is left unresolved.

The system has been evaluated in two different phases. The purpose of the first experiment was to compute the relative performance of CogNIAC to Hobbs approach–a convenient benchmark that allows indirect comparison to other algorithms. The individual rules were also evaluated. The experiment was done on a text with 298 third person pronouns. The Hobbs approach achieved correct result in 78.8% (235 out of 298 cases); whereas CogNIAC achieved correct result in 77.9% (232 out of 298 cases) and the high-precision version of CogNIAC scored a precision of 92% (190 out of 206 cases). The first four rules perform quite well at 96% precision (148 out of 154 cases) and 50% recall (148 out of 298 cases). When it includes the rules five and six then the precision is 92% (190 out of 206 cases) and recall is 64% (190 out of 298 cases). The last two rules, seven (42%) and eight (48%) performed quite badly compared to other rules.

Centering Theory (1997): Centering theory ([127], [128]) is based on focus of attention, choice of referring expression and perceived coherence of utterances in a discourse segment. The theoretical foundation of this work is the concept of the theory of local focusing of Sidner [88]. The main idea of the theory is that, each discourse segment exhibiting a coherence among the utterance is the local coherence which is known as a center; and it changes within the discourse segments. The changes of focus impose certain constraints on the use of referring expressions and in particular, on the use of pronouns. Grosz et. al. [127] formulated the rules to choose the centers and the rules of such transition relations across pairs of utterances based on grammatical role. To understand the algorithm (Algorithm 1) consider the following terms and constraints:
Utterance: A sentence in the context of a discourse.
Discourse segment: Sequence of utterances.
Center: Any entity that is focussed or referred in the discourse.

Forward looking centers $C_f(U_n)$: An utterance U_n is assigned a set of potential next centers $C_f(U_n)$. Where $C_f(U_n)$ is ordered according to their discourse salience (subject, direct object, indirect object, last elements, least salience).

Backward looking center $C_b(U_n)$: An utterance U_n (other than the first) is assigned a single center $C_b(U_n)$. The backward looking center $C_b(U_n)$ is a member of the set $C_f(U_n)$

Prominent center $C_p(U_n)$: is the highest ranking center in $C_f(U_n)$.

According to Grosz et. al. [127] there are three types of transition relations across the pairs of utterances.

Center Continuation: $C_b(U_{n+1}) = C_b(U_n)$ i.e. the backward-looking center of the utterance U_{n+1} is the same as the backward-looking center in the utterance U_n and this entity is the preferred center of $C_f(U_{n+1})$.

Center Retaining: $C_b(U_{n+1}) = C_b(U_n)$ but this entity is not the most highly ranked element in $C_f(U_{n+1})$.

Center Shifting: $C_b(U_{n+1}) \neq C_b(U_n)$

The basic constraints on center realization and center movement is given by the following two rules:

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

Rule 2: Transition states are ordered, i.e. Continuation is preferred over Retaining and Retaining is preferred over Shifting (Continue > Retain > Shift)

```
Generate $C_b$ and $C_f$ assignments for all possible reference assignments
Filter by constraints (syntactic co-reference, selectional restrictions,...)
Rank by preference among transition orderings
```

Algorithm 1: Centering Algorithm

To understand the theory with examples, consider the discourses (Table 1.4) illustrated by Mitkov [8]. The backward-looking centers, forward-looking centers and the transition relation and the utterances are shown in Table 1.5. Note that the backward-looking center is not assigned in the case of the first utterance in a discourse.

Table 1.4: Discourses A and B

Discourse A	Discourse B
S1.1: John works at Barclays Bank.	S1.1: John works at Barclays Bank.
S1.2: He works with Lisa.	S1.2: He works with Lisa.
S1.3: John is going to marry Lisa.	S1.3: John is going to marry Lisa.
S1.4: Lisa has known him for two years.	S1.5: She has known John for two years.

Table 1.5: Transition Relation

Sentence	C_f	C_b	Transition
S1.1 John works at Barclays Bank	{John, Barclays Bank}	{ }	
S1.2 He works with Lisa	{John, Lisa}	John	
S1.3 John is going to marry Lisa	{John, Lisa}	John	Continuation
S1.4 Lisa has known him for two years	{Lisa, John}	John	Retaining
S1.5 She has known him for two years	{Lisa, John}	Lisa	Shifting

Clearly, in the Table 1.5 shows that, sentence S1.3 exhibits center Continuation and in sentence S1.4 it shows the center Retaining but in S1.5 have a center Shift.

Ruslan Mitkov's (1994-2010): An extensive research in anaphora resolution is done by Mitkov ([8], [89], [94], [129], [130], [131], [132], [133], [134]). Several issues in anaphora resolution have been encountered in his research along with the implementation of a system known as MARS. The initial version of the system was reported in [131]. The study shows that anaphora can be resolved quite successfully (at least in a specific genre) without any sophisticated linguistic knowledge or even without parsing. The result showed that just using the basic set of factors the performance was good. Finally, the fully automatic version of knowledge-poor pronoun resolution method [133] referred to as MARS was reported. The main feature of the approach is to avoid the complex syntactic, semantic and discourse analysis. Instead, a set of preferences known as antecedent indicators is used. The algorithm operates on a text pre-processed with POS tagger and an NP extractor. Next, the detected noun phrases are filtered by the gender and number agreement. Collective nouns (e.g. government, team, parliament, organization, etc.), which do not agree in number with their antecedents are considered separately. The antecedent indicators are applied to all NPs filtered by gender and number agreement. The antecedent indicators can have either a boosting effect by applying positive score or an impeding effect by applying the negative score. The boosting and impeding indicators are shown in the Table 1.6.

Table 1.6: Boosting and Impeding factors

Boosting/Impeding Factor	Score
First noun phrases	+1
Indicating verbs	+1
Lexical reiteration	+2
Section heading preference	+1
Collocation match	+2
Immediate reference	+2
Sequential instructions	+2
Term preference	+1
Indefiniteness	-1
Prepositional noun phrases	-1

The algorithm of the system is described in Algorithm 2.

```
1 Consider the current sentence and the two preceding sentences (if available) and
   take the noun phrases only to the left of the anaphora.
2 Filter the selected noun phrases based on gender and number agreement with the
   pronominal anaphora and group them as a set of potential candidates.
3 Assign the score to each potential candidate based on the antecedent indicators
   defined in Table 1.6 and propose the candidate with the highest aggregate. If
   more than one candidate has an equal score, propose the candidate with the
   higher score for immediate reference. If immediate reference does not hold, then
   choose the candidate with higher scores for collocational match (if available). If
   still not resolved, then select the candidate with higher scores for indicating
   verbs (if available) otherwise choose the most recent candidate.
```

Algorithm 2: The Knowledge-poor approach

The knowledge-poor approach was evaluated for English texts from different technical manuals (Minolta Photocopier, Portable StyleWriter (PSW), Alba Twin Speed Video

Recorder, Seagate Medalist Hard Drive, Haynes Car Manual, Sony Video Recorder) which contained a total of 223 anaphoric pronouns. The input text was automatically pre-processed (POS tagging, NP identification) and then manually corrected to ensure that the input to the algorithm was correct. The result was compared with the two baseline models which showed the effectiveness of the approach. The first baseline model which selects as antecedent the most recent noun phrases that matches the gender and number of the anaphora. The second baseline model chooses the most recent subject matching the gender and number. The comparison result is shown in the Table 1.7.

Table 1.7: Boosting and Impeding factors

Approach	Result (%)
Knowledge-poor approach	89.7%
Baseline Most Recent NPs	65.9%
Baseline Most Recent Subject	48.6%

Vieira and Poesio (1997-2004): Considerable research on anaphora resolution is also done by Poesio et. al. ([135], [136], [137], [12], [138]). The main focus of their research is on definite descriptions and to resolve references between definite description[1] and antecedents [12]. The work ([137], [12]) is based on the development of a shallow processing system relying on structural information, lexical resources such as WordNet and hand-crafted information or on information that could be acquired automatically from a corpus. From the methodological point of view, a suitable annotation scheme is chosen for classification of definite descriptions. The systems do not use any pre-processing step but make use of the different types of heuristics.

The algorithm operates on a 4-sentence window and processes the text sentence by sentence. Whenever a new sentence is encountered, all mentions are extracted and the segmentation window is updated. The system then identifies all the definite descriptions by the above heuristics. When the system encounters a definite nominal then uses a decision-tree to determine its classification, and attempts to find an antecedent. The

[1] defined as the NPs preceded by the article the

system was tested on manually developed decision tree with annotated text from Penn Treebank I corpus [139] of newspaper articles from the Wall Street Journal. The training corpus was containing approximately 1,000 definite descriptions used for the development of the system and in the test part of approximately 400 definite descriptions. The overall performance of the system is the precision of 76% and recall of 53%; whereas for direct anaphora resolution the precision is 83% and recall is 62%. Identification of discourse-new descriptions were performed with a precision of 72% and recall of 69%.

Later on they used machine learning techniques [140] to find the best combination of local focus and lexical distance features for identifying the anchor in bridging references. This study uses Multi-layer perceptrons (MLPs) with back-propagation and also used a set of features associated with local focus and lexical distance between anaphora-candidate pairs. The lexical distance is found from two sources, Google distance and WordNet distance. The other features used are based on the interaction of global and local focus [141] i.e. local first mention and global first mention. Back propagation algorithm was used to train the learning system by using both positive and a randomly chosen set of negative instances. The recall rate reported from this study is between 84% and 86% (for a small dataset of 58 instances) with slight variations depending on the combination of features used.

Soon, Ng and Lim (2001): This study makes use of a C4.5-based [142] learning approach [98] to co-reference resolution of noun phrases in unrestricted text. The approach does not resolve certain type of noun phrase (e.g., pronouns) but rather resolves general noun phrases. This method requires a relatively small training corpus that has been annotated with co-reference chains of noun phrases. As per training is concerned, the system requires all the markables (“organization”, “person”, “location”, etc.) along with the necessary information of each markable. Later, the markables are analyzed by the co-reference classifier. It consists of tokenization, sentence segmentation, morphological processing, POS tagging, noun phrase identification, named entity recognition, nested noun phrase extraction, and semantic class determination (shown in Figure 1.3, taken from [98]). For the POS tagger, noun phrase identification and named entities the Hidden Markov Model (HMM) and other statistical learning methods ([143], [144]) have been used. A generic feature vectors has been defined to build a learning-based co-reference engine. A decision tree based learning algorithm [142] has been used to learn a classifier based on the feature vectors generated from the training documents. There are twelve components in the feature vector derived from the linguistic as well as non-linguistic sources. The features are described below considering $\boldsymbol{i}$ is the candidate and $\boldsymbol{j}$ is the anaphora:

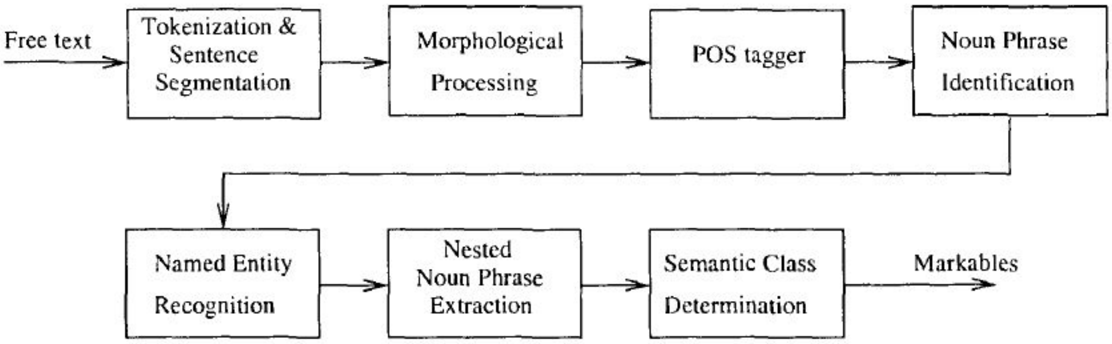

Free text
Tokenization & Sentence Segmentation
Morphological Processing
POS tagger
Noun Phrase Identification
Named Entity Recognition
Nested Noun Phrase Extraction
Semantic Class Determination
Markables

Figure 1.3: System architecture of the pre-processing pipeline used in [98].

Distance Feature - This feature captures the sentence-distance between $\boldsymbol{i}$ and $\boldsymbol{j}$. The possible values are 0, 1, 2, ... where the value 0 represents the same sentence and so on.
i-Pronoun Feature – It returns a Boolean value true or false based on $\boldsymbol{i}$. It returns true if $\boldsymbol{i}$ is a pronoun, else return false. Where, pronouns considered as reflexive pronouns (*himself*, *herself*), personal pronouns (*he*, *him*, *you*), and possessive pronouns (*hers*, *her*).
j-Pronoun Feature - It is same as that of i-Pronoun Feature i.e. return true if $\boldsymbol{j}$ is a pronoun else return false.
String Matching Feature - It returns a Boolean value true if the string of $\boldsymbol{i}$ matches the string of $\boldsymbol{j}$, else false. A pre-processing is done before performing the string matching, i.e. first articles (a, an, the) and demonstrative pronouns (this, these, that, those) are

removed from the strings. Therefore, the string “the license” will match with the string “this license”.

Definite Noun Phrase Feature – It returns a Boolean value; and returns true if $\boldsymbol{j}$ is a definite noun phrase else returns false. For example, “the car” is a definite noun phrase.

Demonstrative Noun Phrase Feature – It returns a Boolean value; and return true if $\boldsymbol{j}$ is a demonstrative noun phrase else return false. A demonstrative noun phrase is defined as that starts with the word this, that, these, or those.

Number Agreement Feature - It returns a Boolean value; and returns true if $\boldsymbol{i}$ and $\boldsymbol{j}$ agree in number (i.e., they are either singular or both plural), else returns false.

Semantic Class Agreement Feature – In the system, the following semantic classes: “female”, “male”, “person”, “organization”, “location”, “date”, “time”, “money”, “percent” and “object” are arranged in ordered [145]. If the semantic classes of markables $\boldsymbol{i}$ and $\boldsymbol{j}$ are same then true is returned, otherwise false. If the semantic class of a markable is not matched with predefined classes, then its class is defined “unknown”. Now if either semantic class is “unknown”, then the head noun strings of both markables are compared and if they match then true is returned; else unknown.

Gender Agreement Feature - If $\boldsymbol{i}$ and $\boldsymbol{j}$ agree in gender, then true else false. But if the gender of either markable $\boldsymbol{i}$ or $\boldsymbol{j}$ is unknown, then the gender agreement feature value is considered as unknown.

Proper-Names Feature – It returns true if $\boldsymbol{i}$ and $\boldsymbol{j}$ are both proper names, else returns false.

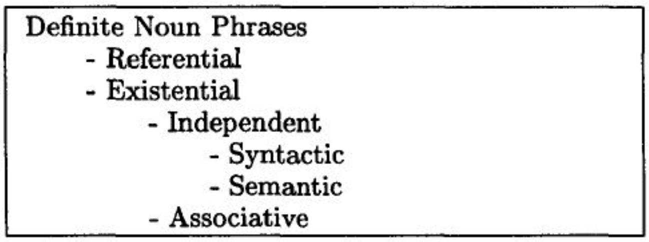
Alias Feature – It returns true if $\boldsymbol{i}$ is an alias of $\boldsymbol{j}$ or vice versa, else returns false.

Appositive Feature – It returns true if $\boldsymbol{j}$ is in opposition to $\boldsymbol{i}$ else returns false.

The resolution algorithm works by generating the feature vector for each of the previous markables starting from the immediate previous one. For each markable $\boldsymbol{j}$, the algorithm considers every markable i before j as a potential antecedent. For each markable pair $\boldsymbol{i}$ and $\boldsymbol{j}$, the generated feature vector is passed to the decision tree classifier. If a co-referring antecedent is found, then the classifier returns true or it processes backward until returns true or there is no remaining markable to test. The system was evaluated on publicly available annotated corpora (MUC-6 and MUC-7) for co-reference resolution. MUC-6 has a standard set of 30 test documents with approximately 12,400 words and for MUC-7 has a test corpus of 20 documents with approximately 19,000 words. The system achieved a recall of 58.6% and a precision of 67.3% and F-measure of 62.6% for MUC-6. On the other hand, for MUC-7 the recall was 56.1%, the precision was 65.5%, and the balanced F-measure was 60.4%.

Ng and Cardie (2002): The research reported in [146] describes a noun phrase co-reference system. It is an extension of the work of Soon et. al. [98]. The extensions over the Soon et. al.'s corpus-based approach are in two different contexts. First was the extension of three extra-linguistic modifications to the machine learning framework. Second, they proposed 26 additional lexical, semantic, and knowledge-based features. Similar to [98] the features are represented as a vector and used decision tree classifier [142]. After training, the decision tree is used by a clustering algorithm to partition the NPs involved in co-referent chain. Similar to [98] words are processed from left to right. On encountering an NP (NP_j), the method compares the NP to each of the preceding NPs, NP_i from right to left. For each pair, a test instance is created as during training and is presented to the co-reference classifier, which returns a value from the interval [0, 1]. The value between 0 and 1 that indicates the probability of the two NPs being co-refereed. To consider as co-referent a threshold value (= 0.5) is considered. The system was evaluated by the same data set as used in [98] and achieved F-measures of 70.4 and 63.4, for the MUC-6 and MUC-7, respectively.

Bean and Riloff (1999, 2004): In [147], Bean and Riloff reported a corpus-based algorithm for identifying the non-anaphoric definite noun phrases automatically. The authors refer to non-anaphoric definite noun phrases as existential NPs [149]. The algorithm [148] uses statistical methods to generate lists of existential noun phrase patterns and noun phrases from a training corpus and lists are then used to recognize existential NPs in new texts. The authors observed that about 50% of definite descriptions had no prior referents [136] and they tried to identify the non-anaphoric definite noun phrases to improve the efficiency and accuracy of co-reference resolution systems. To better understand what makes whether an NP anaphoric, they make use of taxonomy based classification shown in Figure 1.4 (taken from [147]).



Definite Noun Phrases
- Referential
- Existential
    - Independent
        - Syntactic
        - Semantic
    - Associative

Figure 1.4: Definite Noun Phrases Taxonomy

A set of heuristics as explained below has been used for extraction of non-anaphoric NPs.

Syntactic Heuristic: It looks for structural clues of ‘restrictive pre-modification’ and of ‘restrictive post-modification’, respectively. In a noun phrases, the head noun is restrictive pre-modified if a proper-noun is used as a modifier. For example: “the U.S. president”, note that “The president” is ambiguous, but “the U.S. president” is not. Similarly, as an example of ‘restrictive post-modification’, “the president of the U. S.” shows a non-anaphoric definite description.

Sentence one heuristic: If a definite NP appears in the first sentence in a text, then the NP is not anaphoric; since it is unlikely that it would have an antecedent.

Definite-only list: List of some non-anaphoric NPs that never appear in indefinite constructions (e.g. “The F.B.I.” “the contrary”, etc.).

The system was evaluated by the MUC-4 (the fourth Message Understanding Conference, 1992) data. The training set is consisted of 1,600 texts and the test set is consisted of 50 texts and the method achieved 78% recall and 87% precision in identifying such noun phrases.

Cherry and Bergsma (2005): This study proposed an unsupervised Expectation Maximization (EM) [150] based approach [151] to pronoun resolution. The learning is performed from a fixed list of potential antecedents for each pronoun. They showed that the unsupervised learning is possible in this context and the performance of their system is comparable to supervised methods. They have designed the training set as a triplet (p, k, C), where:
p : is the pronoun to be resolved
k : is the context of pronoun p
C: is a candidate list consisting of the candidates up to two sentences backward that precede p
Before passing to EM the necessary modification is done on the triples (p, k, C). The authors treated the third-person pronouns in English based on gender (e.g. masculine (e.g. *he*), feminine (e.g. *she*), or neutral (e.g. *it*)) and number (e.g., singular (e.g. *he*) or plural (e.g. *they*)). Their assumption is that a noun is equally probable to co-refer to any member of a given gender/number category, and reduce each p to a category label accordingly. For example, *he*, *his*, *him* and *himself* are all labelled as masc for masculine pronoun and the others (plural, feminine and neutral) pronouns are handled similarly. Then the system reduces the context term k to p's immediate syntactic context, including only p's syntactic parent, the parent's part of speech, and p's relationship to the parent as determined by a dependency parser. Finally, they ordered the each candidate in C to know how many nouns to "step over" before arriving at a given candidate. The system was evaluated by implementing a fully automatic system and hence automatic pre-processing causes consistent degradation in performance, regardless of the accuracy of the pronoun resolution algorithm. The authors used two training data sets in their experiments, both taken from the Question Answering corpus [154]. The training data are manually labelled with pronoun antecedents. The development data set consists of

3,33,000 pronouns drawn from 31,000 documents. The data set consists of 644 labelled pronouns drawn from 58 documents; 417 are drawn from sentences without quotation marks. The test data consists of 8,90,000 pronouns taken from 50,000 documents. The method achieved the upper bound score (where a correct answer actually appears in the candidate list) is 75% where the base line (previous noun) score is 39%.

Bergsma and Lin's (2006): The approach proposed by Bergsma and Lin [155] is based on syntactic paths. A dependency parse tree is obtained using Minipar [156] parser and then dependency labels between two potentially co-referent are defined. The algorithm first identifies the co-referent paths and non co-referent paths, which are dependency paths that usually lead to co-referential/non co-referential mentions on the two ends. A path instance is considered as likely co-referent if the two pronouns are from the same pronoun group; otherwise it is marked as non-co-referential. Pronouns are partitioned into seven groups based on gender, number, and person. In the determination of path a kind of bootstrapping mechanism has been incorporated. The algorithm makes use of the extension of Bergsma's corpus-based gender and number extraction technique [157]. The system was trained by the data set on the anaphora-annotated portion of the American National Corpus (ANC) used in Bergsma [157], and tested on the MUC-7 test data set; the system achieves an accuracy of 71.6% over the third-person pronouns with nominal antecedents.

Poon and Domingos (2008): Poon and Domingos presented a supervised method [158] for the co-reference resolution. The approach is superior to an earlier unsupervised approach [159] and competitive with the earlier supervised ones. The approach exploits on Markov logic, a powerful representation for joint inference with uncertainty [160] that combines probabilistic graphical models and first-order logic. The model adopts the cluster based approach similar to Haghighi and Klein's [159]. Rather than pair wise comparisons of antecedents and anaphora the method considers all mentions of the same entity as a cluster, and implicitly imposes transitivity. Most importantly, the model leverages apposition and predicate nominals and gives more accurate result, which did not use in the approach in [159]. Poon and Domingos did not predetermine anaphoricity of a mention, but rather combined it into the integrated resolution process and hence the model is inherently joint among mentions and subtasks. The set of simple grammatical features incorporated in the system, including gender and number agreement, head word determination, distance-based salience measure, as well as apposition and predicate nominal relationships. The system was implemented as an extension to the Alchemy system [161]. An experiment was conducted on data set MUC-6, ACE-2004, and ACE Phrase-2 (ACE-2). The dataset MUC-6 consists of 30 documents for testing and 221 for training; ACE-2004 training corpus contains 348 documents where ACE-2 contains a training set and a test set. The evaluation is done by the metric MUC [18]. For the MUC-6 data the MUC scores are: precision = 83.0, recall = 75.8 and F-score = 79.2 and for the ACE data the MUC scores are: precision = 68.4, recall = 68.5 and F-score = 68.4.

Charniak (1998, 2009): Ge et. al. proposed a probabilistic model [162] to resolve third person anaphoric pronouns. They incorporated several anaphora resolution factors to get a single probability which is used to find the antecedent. The system does not use any hand-crafted rules but use the Penn Wall Street Journal Treebank to train their model. The factors the system uses are features such as gender, number, animacy, distance, mention count. Where, distance is the distance between the pronoun and the referring candidate for an antecedent. The greater distance implies the lower probability for a candidate to be the antecedent. Noun phrases that are mentioned more frequently have a higher probability of being the antecedent; the training corpus that is marked with the number of times an NP is mentioned up to each specific point. Later on, Charniak and Elsner [153] have presented an unsupervised machine learning approach. The algorithm is based on Expectation Maximization (EM) to learn virtually all of its parameters in an unsupervised fashion. Actually the work is improved version of Cherry and

Bergsma [151] in several dimensions like the handling of third person Pronouns, distinguish antecedents of non-reflexive pronouns based on syntax and separate preprocessing stage to classify non-anaphoric pronouns. The model ignores the generation of most of the discourse but only generating a pronoun's person, number, and gender features along with the governor (governor is the head of the phrase) of the pronoun and the syntactic relation between the pronoun and the governor. The model first decides if the pronoun is anaphoric based upon a distribution and if the pronoun is anaphoric, then it selects a possible antecedent in the current or two previous sentences. The selection of the antecedent is based upon the distribution p(anaphora|context). Where, the distributions start with uniform values. For example, the gender distributions start with the probability of each gender equal to one-third. From this it follows that, on the first iteration (EM), all antecedents will have the equal probability of generating a pronoun. All the model parameters are learned by EM using the parsed version of the North-American News Corpus [163]. It has about 8,00,000 articles, and 50,00,00,000 words. Finally, the system tested on annotated news articles (MUC-6) and its performance shows that the method is 68% accurate.

Haghighi and Klein (2007-2010): The approach in [164] is a generative, model-based approach in which each of the factors (mentions, entities, etc.) are modularly encapsulated and learned in a primarily unsupervised manner. The authors assume that the frequent co-reference errors in state-of-the art systems are because of the poor models of semantic compatibility [165]. The approach [164] is broadly similar to their earlier approach [78] which addresses this issue. This generative model makes use of a large inventory of distributional entity types, including standard NER types like a PERSON and ORG, as well as more refined types like WEAPON, VEHICLE, etc. The distributions over typical heads, modifiers and governors are learned from large volume of unlabeled data, capturing type-level semantic information for each type. Each entity inherits from a type, but also captures the entity-level semantic information (e.g., "**giant**

may be a likely head for the Microsoft entity but not all ORGs"). A separate log-linear discourse model is used to capture the configuration information. At last, the mention model assembles each textual mention by selecting semantically appropriate words from the entities and types. For evaluation, they have used standard co-reference data sets derived from the ACE corpora ([166], [167]). The evaluation metric MUC [18], B^3 [19], and pairwise F1 has been used. The MUC score gives: precision = 77.0; recall = 66.9; and F-score = 71.6, the B^3 score gives: precision = 55.4; recall = 74.8; and F-score = 63.8 and the pairwise F1 score gives: precision = 60.1; recall = 47.7 and F-score = 53.0. The result shows that it outperforms the previously reported results [165] on an end-to-end co-reference resolution including mention detection.

Raghunathan et. al. (2010): The unsupervised approach [168] based on a multi-pass sieve applies tiers of deterministic co-reference models one at a time from highest to lowest precision. Most of the earlier co-reference resolution models determine if two mentions are co-referent using a single function over a set of constraints or features. But this strategy can lead to incorrect decisions as lower precision features often overwhelm the smaller number of high precision ones. In multi-pass sieve approach, each tier builds on the previous tier's entity cluster output. Also, the model propagates the global information by sharing attributes (e.g., gender and number) across mentions in the same cluster. Hence, it guarantees that stronger features are given precedence over weaker ones and that each decision is made using all of the information available at the time. Since the framework is highly modular: new co-reference modules can be easily plugged in without any effect to the other modules. The modules of the Multi-Pass Sieve consists of seven different passes.

For development and evaluation, the MUC-6 [169] and ACE-2004 [170] corpus have been used. The three evaluation metrics MUC [18], B^3 [19] and pairwise F1 [171] are used. The MUC score gives: precision = 83.7; recall = 74.1; and F-score = 78.6, the B^3 score gives: precision = 88.1; recall = 74.2; and F-score = 80.5 and the pair wise F1 score gives: precision = 80.1; recall = 51.0 and F-score = 62.3. The result shows that the approach outperforms many state-of-the-art supervised and unsupervised models on several standard data sets.

Durrett and Klein (2013): The system reported in [172] is different from the classical rule based co-reference system. It is built on learning-based, mention-synchronous co-reference system that aims to use the simplest possible set of features to tackle the various aspects of co-reference resolution. The system captures the various syntactic, discourse, and semantic phenomena implicitly and a small number of homogeneous feature templates examining shallow properties of mentions. Note that these features are actually more effective than the corresponding hand-engineered ones based on linguistic phenomena. The system first identifies a set of predicted mentions from text annotated with parses and named entity tags and then the mentions are ranked based on features. Finally, it learns to optimize the conditional log likelihood augmented with a parameterized loss function [172]. They used the datasets provided in CoNLL 2011 shared task [173], which is derived from the OntoNotes corpus [174]. The evaluation metrics are considered as MUC [18], B^3 [19] and CEAF [20]. The system gives MUC score is 84.49, B^3 score is 75.65 and CEAF score is 69.89 and the average score is 76.68.

Glen and Hofford (2014): Glen and Hofford describe a new representational scheme [175] for resolving the ambiguous pronouns. This new representational scheme known as ROSS (Representation, Ontology, Structure, Star). The method defined the difficult pronouns as those pronouns for which a level of world or domain knowledge is needed in order to perform anaphoral or other types of resolution. The pronoun resolution algorithm used in the ROSS method starts with entity resolution. It is used in two different ways in support of the pronoun resolution and inference methods: (1) the Star ontology language is used for a specification of object frame classes and for rule-like constructs referred to as behavior classes in the ontology/knowledge base and (2) the formal scheme of the ROSS situation model (also called “instance model”) is used for the specification of meaning representations that represent the semantics of a particular situation. The system is tested with the Winograd schema challenge [176]. The schema is considered as a pair of sentences that differ in one or two words and that contain an ambiguous pronoun that is resolved in opposite ways in the two sentences and also requires the use of world knowledge and reasoning for its resolution.

R¨osiger and Riester (2015): This study examines the effect of prosodic features on a co-reference resolution in spoken discourse [177]. So far, prosodic features have not been taken into account in co-reference resolution. Though there are considerable studies on co-reference resolution in written text but only limited works consider spoken text ([178], [179]). There are differences between written and spoken text with respect to co-reference resolution and obviously the performance typically drops when systems that have been developed for written text are applied on spoken text [180]. The philosophy behind the approach by R¨osiger and Riester is that “there is a tendency for co-referent items, i.e. entities that have already been introduced into the discourse, to be deaccented, as the speaker assumes the entity to be salient in the listener’s discourse model”. They have exploited this by including prominence features in the co-reference resolver. The prosodic information has been used for the purpose of their research results from manual annotations with the annotation scheme GToBI(S) guidelines by Mayer [181]. They mainly make use of *pitch accents* and *prosodic phrasing.*

For the experiment, the German data (DIRNDL corpus[1]) has been considered by adopting the IMS Hot-Coref system [182], which was originally developed for English. These adaptations include number and gender agreement, lemma-based (sub-string) string match. The DIRNDL corpus (50,000 tokens, 3,221 sentences), a radio news corpus annotated with both manual co-reference and manual prosody labels ([182], [183]). The system achieved a CoNLL score (the CoNLL score is defined as the unweighted mean of F1 score of the MUC, B^3 and CEAF) of 47.93, which is considered as a baseline

[1] http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/dirndl.html

result. The study deals with German data, and claimed that, the prosodic features are comparable to other West Germanic languages, like English or Dutch.

Apart from the methods described above some other important works are as follows. Charniak's thesis [107] drew special attention on anaphora in specific domain. Wilks's preference semantics [112] approach used knowledge of individual lexeme meanings in order to resolve sophisticated cases. Kantor [185] investigated ambiguity and different meaning of pronouns, where Grosz [186] concentrated on activatedness (the meaning is very close to that of focus) and Webber [69], Günther and Lehmann's [187], Rolbert [188], Lockman [189], Asher and Wada [190] applied the rule base in their system. Kameyama's algorithm [191] for resolution of nominal anaphora used syntactically incomplete inputs, Tetreault [92] proposed a centering-based pronoun resolution algorithm. Harabagiu and Maiorano [192] used an annotated bilingual English and Romanian corpus to improve the co-reference resolution.

1.4.2 The off-the-shelf Anaphora Resolution Systems

There are several off-the-shelf anaphora and co-reference resolution systems (e.g. MARS, GuiTAR, BART, CoreNLP, ARKref, Reconcile, etc.) are freely accessible to the research community. Some of these are already configured for other languages (e.g. GuiTAR, BART) too. Here some of the widely used systems are described in brief.

GuiTAR (General Tool for Anaphora Resolution[1]) is a highly modular and flexible tool for anaphora resolution and used as an off-the-shelf system [5] implemented in Java. The system takes either XML or raw text as input and produces an intermediate XML (MAS-XML) with the necessary information for anaphora resolution and output is also an XML file annotated with resolution information. This is done based on the GNOME mark-up scheme [11], used to specify the minimal information required by anaphora resolvers. For syntactic information extraction the system uses the LT-XML

[1] http://cswww.essex.ac.uk/Research/nle/GuiTAR/

tools developed by the University of Edinburgh's LTG [193] and Charniak's parser [194]. GuiTAR implementation uses the MARS pronoun resolution algorithm [133], and for definite descriptions it uses the Vieira and Poesio [12] method. The detail description of the system is given in Chapter 4.

BART (Beautiful Anaphora Resolution Toolkit[1]) is an automatic co-reference resolution system [97] which is a product of the project "Exploiting Lexical and Encyclopedic Resources For Entity Disambiguation" at the Johns Hopkins Summer Workshop 2007[2]. The System is based on a statistical approach and implemented in Java. It takes input as XML format and produces output also in XML format with the standoff format of the MMAX2 annotation tool [195]. It uses the parser (YumCha [196] or Charniak's Parser [194]); NER [197] for pre-processing and provides a flexible machine learning model which can be realised by using Weka, Maxent or SVM-light. BART implementation uses the extended version of Soon et. al. [98].

JavaRAP (Resolution of Anaphora Procedure[3]) is the Java implementation of the anaphora resolution system [198] of the Lappin and Leass approach [124]. It only resolves the third person pronouns, lexical anaphora and identifies the pleonastic pronouns. This system also takes the input as a plain text or an XML format and makes use of the Charniak parser [10].

ARKref[4] is a Java-based implementation tool for noun phrase co-reference resolution system [199]. The system is a deterministic rule-based one that uses the syntactic information from a constituent parser along with the semantic information from an entity recognition component. It works on the basis of the Haghighi and Klein algorithm [165].

Reconcile[5] is an automatic noun phrase co-reference resolution system [200] based on supervised learning approach. The system is designed to facilitate the rapid creation

[1] http://www.bart-coref.org/

[2] http://www.clsp.jhu.edu/workshops/07-workshop/exploiting-lexical-encyclopedic-resources-for-entity-disambiguation/

[3] http://aye.comp.nus.edu.sg/∼qiu/NLPTools/JavaRAP.html

[4] http://www.cs.cmu.edu/∼ark/ARKref/

[5] https://www.cs.utah.edu/nlp/reconcile/

of co-reference resolution systems along with easy implementation of new feature sets and approaches to co-reference resolution and empirical evaluation of co-reference resolvers across a variety of benchmark data sets and standard scoring metrics. It abstracts the basic architecture of most contemporary supervised learning based co-reference resolution systems (e.g. Soon et. al. [98], Ng and Cardie [146], Bengtson and Roth [201]) and achieves performance compatible to the state-of-the-art.

1.4.3 Research in Indian Languages

Research in anaphora resolution in Indian languages is very limited. India is a multilingual country with twenty-two official languages. But most of the Indian languages do not have the basic resources, like, POS tagger, chunkier, parser and other necessary sophisticated pre-processing tools. Still, some works on anaphora have been reported on some languages like Hindi, Bengali, Tamil and Malayalam. The earliest system in Indian language is Vasisth which was a rule based multilingual anaphora resolution platform developed by Sobha et al. ([202], [203]). Later on, the authors had exploited the morphological richness of Malayalam and Hindi. Most of the works are done by using the existing available approaches in English. Prasad and Strube [204], Uppalapu et al. [205] and Dakwale et al. [206] have presented different approaches using the Centering theory for Hindi and Dutta et al. [207] have implanted Hobb's algorithm for Hindi anaphora resolution. Murthy et al. [208] has presented a salience factor based approach on Tamil and Sobha et al. [209] presented a salience factor based with limited shallow parsing of text. Akilandeswari et al. [210] used CRFs for resolution of the third person pronoun where Balaji et al. [211] used two stage bootstrapping resolution approach for Tamil.

SemEval-2010 [90] boosted the researchers in co-reference resolution. The SemEval-2010 task was of special interest for us in the following respect: it is the multiple languages co-reference resolution task in multiple languages (six languages such as Catalan, Dutch, English, German, Italian and Spanish). Besides the well-studied languages, the under-resourced languages also took part in the anaphora resolution procedure. The similar event (ICON-2011 NLP tool contest [14] on anaphora resolution) took place for Indic language. The NLP tool contest on anaphora resolution addresses three Indic languages (Hindi, Bengali and Tamil) and the organizer also provided the data for anaphora resolution. The contest was conducted as a part of ICON 2011 [14]. Among the participants, four groups submitted their systems for Bengali. Teams, from ISI, Kolkata [22] used a rule based algorithm. The team from IIT, Kharagpur [26] tried a two-stage approach of identifying the markables using CRFs and identification of links using a decision tree approach whereas, the team from IIIT, Hyderabad [212] has used a hybrid approach i.e. machine-learning approach with the rule based approach. A recent generic work by Sobha et. al. [213] is evaluated for many languages such as Hindi, Bengali, Tamil, and Malayalam. They have used the morphological richness of these languages, limited shallow parsed information along with CRFs to resolve the antecedents.

1.4.4 Work in Bengali Language

Bengali (also known as Bangla) is the second most popular language in Indian subcontinent. It is also the national language of Bangladesh. It is the 7^{th} most popular spoken language in the world [1]. But little computational linguistics research has been done in Bengali so far. As mentioned earlier, the main problem in anaphora resolution in Bengali is the resource constraint, i.e. the pre-processing tools for anaphora resolution are not available or though some tools available, but they do not give reliable accuracy.

Among the very few studies available for analysis of Bengali pronouns, the research reported in [23] and [24] are worth mentioning from the linguistic point of view. Recently, ICON 2011 [14] NLP tool contest on anaphora resolution in three Indic languages included Bengali. The computational effort for Anaphora Resolution in Bengali was taken by Sikdar et al. [25] work. Later, an effort was taken to customize BART [97], which is

[1] http://www.infoplease.com/ipa/A0775272.html

originally designed for English.

The recent study reported in [214] focuses on the feature selection in anaphora resolution for Bengali using multi-objective approach. The technique is grounded on the principle of differential evolution based multi-objective optimization. In this implementation, BART system [97] has been customized and the implementation was evaluated using ICON 2011 [14] NLP tool contest data.

1.5 Contribution of this Thesis

This thesis addresses the problem of pronominal anaphora resolution (PAR) in Bengali. While dealing with the problem, existing works are studied in details. This thesis documents the state of the art in the field of PAR and this shows that though many researchers have considered this problem in English, very little initiative is taken for Indian languages. This situation motivates the subsequent research embodied in this thesis which can be considered as a pioneering research effort from many aspects as described below.

Designing of a PAR system essentially demands a systematic study of Bengali pronouns. Though a few studies have been reported before in the linguistics viewpoint, but the study of pronouns in the computational view point not had been done well. This thesis presents an elaborated study of Bengali pronouns. The study is based on a large corpus and leaves a significant contribution to the future NLP research in Bengali. Several important aspects are brought out and many of which essentially help in configuring a robust PAR system.

Success of an advanced NLP task like PAR depends on several other NLP tools like morphological analyser, POS tagger, chunkier, named entity recognizer (NER), parser, etc. Like most of the Indic languages, Bengali also suffer from NLP resource scarcity and hence, this thesis puts a great deal of effort in developing basic NLP tools needed for designing a PAR system in Bengali. In developing such tools, several off-the-shelf systems working well in English and other languages have been studied and reused whenever possible. A few tools have been designed from the scratch. Many of such tools are now made available online [1] in order to facilitate future NLP research in Bengali.

The next contribution of this thesis is to study the existing PAR systems as so to reconfigure them for Bengali. This is a pioneering attempt for any Indic languages as though there were a few studies on anaphora resolution, but no attempt was made to check whether any of the existing PAR systems is suitable for dealing with Indian languages. In this concern, the system GuiTAR was studied in details and its language-dependent components are identified and suitably redesigned for Bengali. A GuiTAR-based PAR in Bengali could be considered as a significant contribution of this thesis.

Next, the thesis presents a novel PAR approach. In most of PAR approaches, the dominant trend is to look for the correct antecedent after a pronoun is encountered. Unlike this trend the PAR system presented in this work follows a unique approach. On encountering a noun phrase (which could be a possible antecedent) this approach emits a list of permissible pronouns (note that a noun cannot associate with all possible pronouns). Next, when a pronoun is actually encountered, list(s) containing this pronoun is searched. The noun phrase that emitted this list emerges as the right antecedent. If the pronoun does exist in more than one list, a conflict resolution protocol is executed to find out the right antecedent.

Apart from using some rule-based methods, the thesis also attempts to explore the potential of some well-known machine learning algorithms. The Maximum entropy based algorithm has been designed for identification of honorific information. Later, conditional random fields (CRF) and support vector machines (SVM) have been used for classification tasks. As a whole this thesis presents notable experiences of using rule-based and statistical approaches for different tasks and sometimes hybrid of these approaches.

[1] http://www.isical.ac.in/∼utpal/resources.php

After designing a basic PAR system for Bengali, the thesis further contributes in studying several advance issues relevant for developing an improved PAR, in particular and advanced NLP tools, in general. Subject and object drops have been studied. Use of one-expressions in Bengali has been studied and then computational means for automatic analysis of one-expressions are developed. For computational linguistics of Bengali, all such studies (i.e. drop detection, study of one-expression or one-anaphora) are probably done for the first time. Use of such techniques in improving machine translation has also touched upon in this thesis.

The thesis also contributes in evaluating the PAR approaches. Several evaluation metrics are in use for measuring efficiency of a PAR system. Five such commonly used metrics namely, MUC, B-Cube, CEAFM, CEAFE and BLANC are implemented under this thesis work. The publicly available ICON 2011 dataset consisting of nine articles has been used for the evaluation. The dataset is further extended by adding four articles annotated in a format same as of ICON data. Use of such a publicly available data set and commonly used metrics will help the readers to understand the state of the art of the related research and the progress made by this thesis work.

1.6 Organisation of this Thesis

The thesis is organized into seven chapters and the brief outline of the chapters are as follows:

Chapter 1: The first chapter discusses an overview of the general background and the problem setting and motivation to choose PAR in Bengali. This chapter also outlines the previous relevant contribution on the theories and implementations of the anaphora resolution algorithms. It clearly shows the evolution in the resolution strategy, i.e. gradually how the algorithms shifted from the rule based to machine learning based approaches over the time. The literature review also illustrates that the machine learning approaches are the current trend and they have achieved success rates competitive to the earlier traditional rule based approaches. The salient contribution of the thesis is also presented in this chapter.

Chapter 2: This chapter discusses about the existing studies on Bengali pronouns. These studies are mostly in the domain of linguistics and address issues like the pronouns with origin, etymological history, classifications, syntactic and semantic study. A corpus based analysis of Bengali pronouns is presented in this chapter. The morphological structure of Bengali pronouns is also explored. Next, various features and exceptions of pronouns are recorded with suitable examples and utility of these features and exceptions is explained in the context of anaphora resolution. The two corpora (TDIL and FIRE) have been used for this study. This chapter also presents the frequency of pronouns and the coverage in the corpus. The well known LL test is adopted for the significance test. Finally, the relative rank list of pronouns is computed and the list shows that the personal pronouns dominate in the language. Several important aspects are coming out of this chapter which are essentially helping in configuring a robust anaphora resolution system.

Chapter 3: This chapter attempts to overcome the basic resource scarcity for the anaphora resolution. The data used for anaphora resolution must be annotated with at least POS tags, NE tags, chunking information and during resolution process the morphological information is required. This chapter deals with developing these basic NLP tools like POS tagger, NER and morphological analyser. The Stanford University POS tagger [62] has been retrained for the Bengali language. The tagger originally was developed for English language based on Maximum Entropy principal and we have trained it for Bengali language. To train the system, annotated data has been taken from various sources ([14], [75]) and IIIT Hyderabad [241]. The system achieved about 84% accuracy. Next, the chapter concentrates on developing a Morphological analyser. Our approach is a rule based one, implemented based on the principle of finite automata. The automata are used for suffix matching with a predefined list of suffixes. The suffix list is collected from the training data of FIRE-MET[1] test data. The inflections of numerous Bengali words have irregular basis and these are encountered by using a map corresponding to their root words. Finally, the chapter develops a rule-based NER, and it is realised by using untagged data. The rules are formulated with the context information of a named entity. The honorific context, surname context, middle name context, relational term context, context for location, distance context, etc. are used.

Chapter 4: The central goal of this chapter is to re-configure an existing system GuiTAR (originally developed for English) for Bengali. Initially, the system GuiTAR has been studied in details and the language specific issues are identified to re-configure it for Bengali. Such issues are broadly classified into two categories: pre-processing steps and language dependent parts. In pre-processing pipeline GuiTAR creates the MAS-XML which contains all the necessary annotations for anaphora resolution. Here we resolved the issue by creating the MAS-XML externally from our tagged data [14]. In the original version of GuiTAR, the gender agreement has been used but this agreement does not valid in Bengali. Instead the honorific agreement is used in Bengali. This is resolved by removing the gender agreement and introducing the honorific agreement in the implementation. Finally, the system is evaluated by the publicly available data [14] which is further extended at our lab. An error analysis is also done to identify the weaknesses of the configured system.

Chapter 5: This chapter presents a novel approach for pronominal anaphora resolution (PAR) in Bengali. In most of the existing PAR approaches, the dominant trend is to look for the right antecedent after a pronoun is encountered. Unlike this trend the PAR system presented in this chapter follows a unique approach. On encountering a noun phrase (which could be a possible antecedent) this approach emits a list of permissible pronouns (note that a noun cannot associate with all possible pronouns). Next, when a pronoun is actually encountered, list(s) containing this pronoun is searched. The noun

[1] http://www.isical.ac.in/∼fire/morpho/MET.html

phrase that emitted this list emerges as the right antecedent. If the pronoun does exist in more than one list, a conflict resolution protocol is executed to find out the right antecedent. A set of rules is used along with the emitting approach. The rules are also evaluated for their degree of confidence. The system is evaluated with the publicly available data set and comparison is done with the GuiTAR-based implementation. An error analysis is done and some issues are identified for future improvement.

Chapter 6: This chapter discusses mainly three advanced research issues namely, analysis of role of verb, one-expressions, and pro-drop with respect to the Bengali language. The issues are not only important in anaphora resolution perspective, but also important for several other NLP applications. Since there is no significant computational work on these issues, our work provides a number of important insights in the context of computational linguistics of Bengali. At first, a corpus based analysis is provided that investigates quantitative measures of pro-drop and one-expressions in the text. Necessary tools and resources for conducting this research are mentioned. Development of computational approaches is also presented. Finally, the influence of such issues on the development of the robust anaphora resolution system is considered. In summary, this chapter enlightens some future research areas in the domain of discourse analysis in Bengali.

Chapter 7: This chapter presents concluding remarks. It starts with stating the goals originally set before starting the work embodied in this thesis. Given this set of goals, the achievements of this research are highlighted in this chapter. Finally, the chapter points out the issues that may be addressed in further extension of this research.

Chapter 2

Corpus-based Study of Bengali Pronouns

2.1 Introduction

Most of the Indian Languages are inflectional in nature and Bengali is one such example of a highly inflected language. Unlike English or similar languages numerous pronouns exist in Bengali because of the inflectional variations. Almost all the pronouns are inflected through case marking and/or other suffixing. These inflections sometime not only give syntactic information, but also give the semantic information. Generally, morphological analyser is used for word-level morpho-syntactic analysis. Although some studies in Bengali morphology are available ([215], [216], [217], [218]), but none of these performs well in case of pronouns. Because most of the inflections of pronouns show irregular (formation of inflection does not follow specific rule) basis and obviously a separate morpho-syntactic analysis is needed for pronouns. Another aspect is the presence of lot of exceptions for Bengali pronouns at syntax as well as functional level. In many languages the pronoun referent agreements are on the basis of gender, number, etc., but in Bengali the gender agreement is not applicable. In Bengali there is no distinction of pronouns based on gender information; however, depending upon honorific information different pronouns are used. Also Bengali language has the diglossic property, i.e. two distinct varieties of a language are found in spoken as well as written form within the same community and formally known as *sadhu bhasa* (*sadhu* language) and *cholti bhasa* (*chalti* language). In some cases the use of pronouns also depends on its diglossic property, i.e. separate sets of pronouns are used for *sadhu bhasa* (*sadhu* language) and *cholti bhasa* (*chalti* language). So, such analysis of pronouns is very helpful for several NLP tasks, especially in the pronominal anaphora resolution. This chapter also gives a statistical analysis along with the morpho-syntactic study.

2.2 Existing Works on Bengali Pronoun

The pronoun has been studied over the years and the research has mainly found its application in the linguistics discipline. The most significant work on Bengali pronouns found from Chatterji [219]. Chatterji gives a list of pronoun roots with their origin and classification and provides a comparative study with pronouns available in other Aryan languages. But for the lack of a corpus and computing facility at earlier days it was not possible for any exhaustive statistical measure. Sen [220] provides an etymological history of Bengali pronouns and Sarkar et al. [221] classified them and explored their uses in texts. Chaki [222] discussed about the classes of pronouns, nouns used as pronouns and some pseudo pronouns found in Bengali. Dash [223] presented a corpus based analysis of Bengali pronouns. This work explored the morphological structure of Bengali pronouns. The morphological structures of Bengali words (including pronouns) are also analyzed by Bhattacharya et al. [224]. Thompson [225] has explained many syntactic as well as semantic features in her work. Some of these features were not explored earlier.

2.3 Exceptions in Bengali pronouns

There are lots of exceptional features and ambiguities in Bengali pronouns at syntactic as well as semantic level. These features sometimes give valuable information in language processing or sometimes may be the cause of some processing errors. Some of the important features and exceptions are described below.

2.3.1 Highly Inflectional

Bengali is morphologically rich and highly inflected language and hence almost all the words including pronouns are inflected. Most of the inflections in Bengali are generated by the suffixing (mainly) or prefixing (sometimes) of the root words. Hence, it results in a large number of inflectional pronouns. For example, in our corpus based study, we have noticed that eighty-seven inflected form of the pronoun সে/*se* (he or she) exists. All such inflected forms may not be pronouns (i.e. the POS tags of many inflected forms may not be pronoun) but in several cases the inflected forms may be anaphoric.

2.3.2 Prefix Property

All the Bengali words are inflected in the form of either prefix or suffix. But the interesting observation is that prefix-ion (i.e. inflection in the prefix) is not seen for Bengali pronoun. Only the inflection seen in the suffix position for pronouns. Therefore computationally, we can assume that a root is the prefix of all its inflectional forms and is known as the prefix property. This property helps us to find all other inflected pronouns for a given root pronoun from the corpus.

However, many of the inflected forms of pronouns are irregular, i.e. inflected pronouns are not just the concatenation of a root pronoun and some suffix. Handling of such inflections using uniform rules is difficult. For example, the root pronoun আমি/*Ami* (I) having the irregular inflectional forms like আমার/*AmAra* (my), আমাকে/*AmAke* (to me), আমাদের/*AmAdera* (our) etc. In such cases the root pronoun (আমি) is not exactly the prefix of the inflected forms (আমার/*AmAra* (my), আমাকে/*AmAke* (to me), আমাদের/*AmAdera* (our)). For handling these cases we have used some heuristic along with the prefix property to find all the inflected pronouns.

2.3.3 Ambiguity in Bengali Pronouns

As said earlier, there are so many ambiguities and exceptions in Bengali pronoun. The ambiguities of Bengali pronouns are explained at different levels. Broadly, we define the ambiguities as of two types namely, syntax level ambiguity and semantic level ambiguity.

2.3.3.1 Syntax level ambiguity

The syntax level ambiguity arises when a word may appear in the text as a pronoun (PRP) or as other parts-of-speech. In our observation word may appear as a pronoun (PRP), a conjunction (CC) or a noun (NN), etc. The ambiguity can be resolved by using the context information. For example, in a discourse the POS of word তার/*tAra* may be PRP or NN. Consider the following sentences:

- তার নাম কি ?/ *tAra nAma ki* ? (What is his name?).
- ধাতব তার তড়িতের সুপরিবাহি/ *dhAtaba tAra tariter suparibAhi* (Metallic wire is a good conductor of electricity).

In the first sentence the word তার/*tAra* is PRP and meaning is *his* (or *her*). But in the second sentence, the word তার/*tAra* is a NN and its meaning is *wire*. Similarly, the word ও/*o* may appear either as PRP or CC. In the corpus[1] we have identified six such ambiguous words which are used not only as pronoun, but also as some other POS. Examples of such instances are ও/*o*, তার/*tAra*, এক/*eka*, আপন/*Apana*, তাই/*tAi*, যাতে/*yAte*. Also note that some inflected pronouns change its meaning as well as POS. For example, নিজস্বতা/*nijasbatA* (personality), আমিত্ব/*Amitba* (egotism), আত্মস্থ/*Atmastha* (realization), etc.

[1] http://tdil.mit.gov.in/

2.3.3.2 Semantic level ambiguity

Semantic level ambiguity arises when a word appears in the text as a PRP but its semantic depends on the use of the word in the sentence. Consider the following examples:

- আমি আপনি যাব /*Ami Apani yAba* (I will go by myself)

- আপনি কখন যাবেন ?/*Apani kakhana yAbena*? (When will you go?)

In both sentences, the POS tag of word আপনি/*Apani* is PRP. But in the first case the semantic of আপনি/*Apani* is **myself/self**, whereas in the second case the semantic of আপনি/*Apani* is the **honorific form of you**. Similarly, in the example, তুমি কি খাবে/*tumi ki khAbe* have different semantics (**will you eat** or **which food will you eat**) and semantic can be determined from the discourse knowledge. In this case, the ambiguity comes for the use interrogative pronoun কি/*ki* (what).

2.3.4 Reduplicated Pronouns

The definition of reduplication is the repetition of the smallest linguistic unit partially or completely, i.e. repetition of phoneme, morpheme, word, phrase, clause or the utterance as a whole and it gives a different meaning in syntax as well as semantic level. Pronoun reduplication can have significant impact on syntactic and semantic interpretation. In our corpus-based study, we have found more than 105 instances of re-duplicated pronouns, but 82 of them have a high frequency. Examples of some instances are, কে কে /*ke ke* (reduplicated form of who), কি কি/*ki ki* (reduplicated form of what), কোন কোন/*kona kona* (reduplicated form of which), কেউ কেউ/*keu keu* (some of them), কোথাও কোথাও/*kothAo kothAo* (somewhere) etc. Sometimes reduplicated pronouns are used for giving emphasis. They may change number information from singular to plural in a sentence. Consider the following examples:

- তুমি কি খাবে ?/ *tumi ki khAbe*? (Which food item would you eat?).

- তুমি কি কি খাবে ?/ *tumi ki ki khAbe*? (Which food items would you eat?).

In these examples, the second sentence has the reduplicated use of the pronoun কি/*ki* (what). Note that the single occurrence of pronoun (specially the interrogative pronouns) is singular whereas the reduplicated pronoun changes its number to plural. For example, the pronoun কে/*ke* (who) is singular, but its reduplicated form is কে কে/ *ke ke* (reduplicated form of who) refers plurality. In some cases (especially the third person personal pronoun) the reduplication changes semantic from definite to indefinite form. Examples of such instances are, কেউ কেউ/*keu keu* (some people), কোনো কোনো /*kono kono* (sometime), কোথাও কোথাও/*kothAo kothAo* (somewhere) etc.

2.3.5 Co-occurring (Correlative) Pronouns

Some Bengali pronoun pairs appear in the language and they maintain a correlation among them. Consider the following examples:

- জীভ দিয়েছেন যিনি, আহার দেবেন তিনি/ *jiba diyeChena yini , AhAra debena tini* (Who has given us mouth will give us food).

- যে ঈশ্বরে বিশ্বাস করে, সে সুখী হয় / *ye Ishbare bishbAsa kare, se sukhi hay* (One who believes in God becomes happy).

Observe that in the first sentence the co-relative pronoun of যিনি/*yini* (honorific form of who) is তিনি/*tini* (honorific form of he or she) and in the second sentence, the co-relative pronoun of যে/*ye* (who) is সে/*se* (he or she). Our observation shows that these pronouns pairs are always co-referent and the morphological constructions of these pairs are same. From our corpus based study, we have found 47 such pairs. The interesting observation is that when a pronoun is immediately followed by its correlative pair then its semantics is changed. Consider the following examples:

- যাকে তাকে দিয়ে একাজ হবে না/ *yAke tAke diye ekAja habe nA* (This work can't be done by an arbitrary person).

- উনি যে সে ব্যক্তি নয়/ *uni ye se byakti nay* (He is not an ordinary person).

In the first sentence, the meaning of co-occurring pair যাকে তাকে/*yAke tAke* is an **arbitrary person** and in the second sentence, the meaning of যে সে/*ye se* is a **significant** person.

2.3.6 Morphological suffixes in Bengali Pronouns

Suffixing is a very common feature and has semantic impact on the words. They can change the meaning of the words or they can change the syntactic word class and so on. The morphological suffixes can be grouped primarily into three classes based on their function. They are Classifiers, Case Markers and Emphatic Markers. In case of pronouns these suffixes give the valuable semantic and grammatical information regarding their referents. These are very important clues that are useful in several NLP applications, especially in anaphora resolution, machine translation, etc.

For example, in case of number agreement the classifiers টা/ *TA*, গাছা/*gAChA*, খানা/*khAnA*, জন/*jana* are used for singularity and রা/*rA*, দের/*dera*, গুলো/*gulo* are used for plural identification for noun and pronoun. Sometimes classifiers give information regarding definiteness (সেইটা/*seiTA*), indefiniteness (আর পাঁচটা/*Ara pA.NchaTA*), honorific information (লোকটা/*lokaTA* implies a non honorific person whereas লোকটি/*lokaTi* implies a honorific person), etc.

There are some exceptional nature of Bengali pronoun suffixes compared to other parts-of-speech. The most notable feature is that, a singular stem can use a plural suffix, e.g. আমারগুলো/*AmAragulo* (of mine) while the plural stem can use a singular suffix, e.g. তাদেরটা/ *tAderaTA* (of them). In such cases the semantic information of the pronoun is also changed. In the above example, the pronoun আমার/*AmAra* (my) is the singular possessive pronoun whereas after affixing of the plural suffix (গুলো/*gulo*), - আমারগুলো/*AmAragulo* represents some (plural) inanimate items (belonging to me). Similarly the pronoun তাদের/*tAdera* (their) is the third person plural, whereas after affixing of the singular suffix

টা/*TA*, - তাদেরটা/*tAderaTA* refers to an inanimate singular item (belonging to them). But such types of changes are not always straight forward, some exceptions are also noted. Consider the following example:

- আমি আমারটা বুঝে নেব/ *Ami AmAraTA bujhe neba* (I will take care of my business).

In this example, the meaning of the pronoun আমারটা/*AmAraTA* (of me) is **my business**.

2.3.7 Postposition in Bengali Pronouns

Postpositions in Bengali are like prepositions in English, but they are placed after the words. Postpositions perform functions similar to inflectional markers. Sometimes postpositions are themselves inflected by markers such as রা/*rA*, এরা/*erA*, দের/*dera*, কে/*ke*, etc. From the semantic point of view the postpositions provide significant information regarding spatial, temporal, comparative, content, causal, manner related, locative and other aspects of the pronoun (and noun) which give important clues for pronominal anaphora resolution. In our context, sometimes postpositions carry a significant amount of semantic information for pronouns. Some pronouns appear in the text as polysemous word (for example: তার/*tAra*, সে/*se*, ও/*o*, etc.). In such cases in order to conclude the semantic nature of pronoun the analysis of its position is helpful. Consider the following examples:

- হলদিয়া পেট্রোকেমিক্যাল প্রকল্প রূপায়নে যাতে আর দেরি না হয় , সরকার সে দিকে নজর রাখবেন/*haladiyA peTrokemikyAla prakalpa rUpAyne yAte Ara deri nA hay , sarakAra se dike najara rAkhabena* (The government will enough pay attention so that implementation of Haldia petrochemical project is no longer delayed).

- যে মানুষ মরতে চলেছে তার পক্ষে সবই সম্ভব/*ye mAnuSha marate chaleChe tAra pakShe sabai sambhaba* (A suicide bomber can do anything).

In the first example the postposition of the pronoun সে/*se* is দিকে/*dike* and it gives the semantic about সে/*se* that it cannot be anaphoric. In the second example, the postposition of তার/*tAra* is পক্ষে/*pakShe* and it gives the semantic about তার/*tAra* that it is always anaphoric and its referent should be a person.

2.3.8 Gender and Bengali Pronouns

The Bengali language doesn't have gender-specific pronouns, i.e. pronouns do not distinguish gender. Same set of pronouns refers to both male as well as female. For example, there is no distinction between *he* and *she* (*him* and *her*) in Bengali, both refer to the same pronoun as সে/*se*. Bengali verbs also don't change based on the gender of the person, but it does in some other Indo-European languages such as Hindi. It should be noted that the Bengali language has the gender-specific nouns like অভিনেতা/*abhinetA* (actor) , অভিনেত্রী/*abhinetrI*(actress); যুবক/*yubaka* (young man) , যুবতী/*yubatI*(young woman), etc.

2.3.9 Honorificity of Bengali Pronouns

The honorific agreement is important in Bengali and it is used for personal nouns and pronouns. This feature distinguishes people on the basis of their social status. Three types of honorificity exist in written as well as in spoken form. In Bengali, separate set of dedicated pronouns exists in each category. The highest degree of honorificity normally refers to people of high social status like doctors, teachers, lawyers, professors, political men, etc. or parents, grand parents, senior people of the family or society, etc. or sometimes unknown respected person. The dedicated pronoun set for this category is {আপনি/*aApani*, তিনি/*tini*, উনি/*uni*, etc.}. The next level of honorificity is the neutral form, it refers to members of a family who are very close to each other, children or younger members of family, or people within a peer group. The pronoun set for this category is {তুমি/*tumi*, তার/*tAra*, তাদের/*tAdera*, etc.}. The lowest level of honorificity normally refers to very close friends, very close relations (who are younger, e.g. son, daughter, etc.) or the people presumed to be of inferior social status like, housemaids, rickshaw-pullers, and other menial service workers. The pronoun set for this category is {তুই/*tui*, তোর/*tora*, তোকে/*toke*, etc.}.

2.3.10 Pro-drop in Bengali Pronoun

Bengali is a pro-drop (pronoun drop) [225] language, but in general pro-drops imply the subject or object drops. The conjugated verb forms give a clear clue of the drop information. The pro-drops play a vital role in several NLP applications such as anaphora resolution, co-reference resolution, machine translation, etc. Since, the pronouns refer to either subject or object and hence the pro-drop has a big impact in anaphora resolution. Especially, in verbal communication the pronoun drop is quite frequent. Consider the following examples:

- চা খাব/*chA khAba* (I will take tea).

- ফুটবল খেলছেন/*phuTabala khelaChena* (He is playing football).

In the first sentence, there is a subject drop or particularly the drop is a pronoun (আমি/আমরা/*Ami/AmarA* (I/we)) and it can be determined by analyzing the verb খাব/*khAba* (shall eat). Similarly, in the second sentence the drop is also a pronoun তিনি/তাঁরা/*tini/tA.rA* (he/she/they).

2.3.11 Noun act as Pronoun in Bengali

Sometimes some nouns also play the role of pronoun in the text. For example, consider the sentences:

- হুজুর একবার এই বান্দাকে আজ্ঞা করুন/ *hujura ekabAra ei bAndAke A njA karuna* (Your majesty, you just order this servant).

- অধমের নাম চরনদাস/*adhamera nAma charanadAsa* (My name is Charandash).

Clearly in the first sentence the nouns হুজুর/*hujura* and বান্দা/*bAndAke* are acting as pronouns. In this sentence, the হুজুর/*hujura* implies *you* and বান্দা/*bAndAke* implies *me.* Similarly, in the second sentence the meaning of অধমের/*adhama* is *my.* We have found nine such nouns in our corpus based study (দাস/*dAsa*, দীন/*dIna*, অধম/*adhama*, বান্দা/*bAndA*, সেবক/*sebaka*, সেবায়েত/*sebAyet*, গরীব/*garIba*, গোলাম/*golAma*, হুজুর/*hujura*). But uses of such nouns are less frequent in news text, daily communication, magazine, etc. Generally, these are found in *sadhu* language, poem, novel and in earlier (early 19^{th} century) literature.

2.3.12 Pronoun adopted from other languages

Bengali has a large vocabulary and a large number of vocabularies adopted from foreign words [225] like Hindi, Arabic, Persian, Turkish, etc. But none of the pronouns from other languages are adopted in Bengali except Hindi. In our corpus-based study some Hindi pronouns are found in Bengali written in Bengali script. The examples of such pronouns are, হামি/*hAmi* (I), হামার/*hAmAr* (my), উসকা/*usakA* (him), য়্যায়সা/*yAysA* (this), উসকো/*usako* (his), তুম/*tuma* (you), etc.

2.3.13 Idiomatic nature of Bengali Pronoun

Sometimes, two or more pronouns together behave like idiom, i.e. it means something other than the literal meaning of its individual pronouns. Consider the following examples:

- কি যা তা বকছ/ *ki yA tA bakaCha* (What are you babbling nonsense).
- যার যেমন তার তেমন ভগববান ঠিক করেই রেখেছেন/*yAra yemana tAra temana bhagababAna Thika karei rekheChena* (God has already determined the perfect fortune).

Clearly in the first sentence the meaning of the phrase কি যা তা/*ki yA tA* is *nonsense* but each individual pronoun has a separate meaning. Similarly, in the second example, the meaning of the phrase যার যেমন তার তেমন/*yAra yemana tAra temana* is *perfect fortune*. Some of these examples are যে যেমন সে তেমন/*ye yemana se temana*, যখন যেমন তখন তেমন/*yakhana yemana takhana temana*, কে কোথাকার কে/*ke kothAkAra ke*, যে যার সে তার/*ye yAra se tAra*, etc. Our observation is that when three or more than three pronouns are collocated in the text, then we notice the idiomatic nature.

2.3.14 The Pronouns used in prosody

Some pronouns in Bengali are dedicated for prosody and rarely used in prose or spoken form. We have found such pronouns like মম/*mama* (my), আমা/*AmA* (to me), মোর/*mora* (my), মোরে/*more* (me), মোরা/*morA* (we), মোদের /*modera* (our), আমরি/*Amari* (my), etc. The occurrence frequencies of such pronouns in the corpus are low.

2.4 Corpus-based Analysis of Bengali Pronouns

In our corpus-based analysis of Bengali pronouns, the Technology Development for Indian Languages (TDIL[1]) corpus and the Forum for Information Retrieval Evaluation (FIRE[2]) news corpus have been used. The TDIL corpus is developed by the Department of Electronics, Govt. of India for Bengali language. This corpus contains texts from the Literature (20%), Fine Arts (5%), Social Sciences (15%), Natural Sciences (15%), Commerce (10%), Mass media (30%), and Translation (05%). Where each category has some sub categories, e.g., Literature includes novels, short stories, essays, etc.; Fine Arts includes paintings, drawings, music, sculpture etc.; Social Science includes Philosophy, History, Education etc.; Natural Science includes Physics, Chemistry, Mathematics, Geography etc.; Mass Media includes newspapers, magazines, posters, notices, advertisements etc.; Commerce includes Accountancy, Banking, etc., and Translation includes all the subjects translated into Bengali. Whereas, the FIRE corpus is the news corpus

[1] http://tdil.mit.gov.in/

[2] http://fire.irsi.res.in/fire/data

collected from a leading Bengali daily newspaper *Anandabazar Patrika* (ABP[1]) for ten years (2001 - 2010). It is the collection of news articles on different domains (political, economic, social, sports, story, science article, notices, advertisements, travel article, literature, etc.). The corpus sizes are shown in the following table (Table 2.1).

Table 2.1: The corpora used in pronoun analysis

Corpus	# Files	# Sentences	# Words	Disk space
TDIL	1,362	334,260	4,429,574	67.7 MB
FIRE	491,149	12,531,364	170,657,863	2.70 GB

2.4.1 Corpus-based Statistics of Bengali Pronouns

In order to explore the statistical analysis of the pronouns in the corpus, we have measured the frequency of root pronouns, frequency of inflected pronouns, frequency of sentences containing pronouns, etc. The detailed relative frequency measure of the pronouns in the corpus is shown in the Table 2.2. The table also shows the comparison of statistics found in TDIL and FIRE corpora.

Table 2.2: Distribution of pronoun in corpora

Corpus	% Sentence containing pronoun	# Pronoun	% Pronoun (among all POS)
TDIL	71.11	560,696	12.65
FIRE	66.9	16,371,904	9.59

At a glance, the result in Table 2.2 shows that the percentages of the pronouns in the corpora are 12.65% and 9.59% respectively, and about 50% (i.e. 50.18% and 49.9% in TDIL and FIRE corpus, respectively) of pronouns are root pronouns in both cases. We have analysed the statistical significance this result[2] (usually 99% or 95% certain). Two common tests of significance are chi-square and log likelihood (LL) [226]. The LL test is adopted to assess the significance of the difference between frequency scores. According

[1] http://www.anandabazar.com/

[2] http://www.lancaster.ac.uk/fss/courses/ling/corpus/blue/l08_4.htm

to Oakes [227] the LL test score must be greater than 3.84 for statistically significant. In our experiment the LL score by Rayson's method [228] is 38,272.19 and hence the difference in the pronoun frequencies in the two corpus is significant.

Dash [223] investigated the Bengali pronouns in a corpus of 3,000,000 words and reported the following statistics. In that corpus, 3% of the word forms are pronouns. Of these pronouns, 75% are inflected with case markers or other suffixes, while 25% of the pronouns were root forms. The number of individual pronoun word forms in the corpus is around 800 and the number of pronoun roots is 65. But our results deviate from this result. We have found the number of root pronouns is 71. Whereas, the number of individual pronouns (including the inflectional forms, re-duplicated, and এক/*ek* (one)) is 1,515. Actually the distribution of pronouns varies from corpus to corpus and depends on many other factors like size, domain of the texts, definition of pronouns, etc. For example, we have considered এক/*ek* (one) as a pronoun (because of its anaphoric nature in several cases) but this was not considered in the earlier study [223].

2.4.2 Rank of Pronouns in corpus

While the words (surface words) are arranged according to their occurrence frequencies in the corpus, result shows that the rank of the word (এ/*e* (this)) that is used as a pronoun is three i.e. just after the two most frequent words (না/*nA* and করে/*kare*) in Bengali. And within the top most frequent 100 words, the number of words that are used as pronouns is significant (Table 2.3). The following Table 2.4 shows the details ranking of pronouns in the corpus.

Table 2.3: Relative rank of pronoun in corpus

Corpus	Top rank of the word that is used as Pronoun	No. of words used as pronouns within the top 100 ranks
TDIL	3	35
FIRE	3	26

Note that we have defined the term *word that is used as pronoun* in the Table 2.3 instead of the pronoun, because some words may appear either as a pronoun or as other POS in the text and it depends on the context of the sentence. Also note that the ranking is done on the basis of the surface words in the corpus.

Table 2.4: Top 10 most frequent pronouns

TDIL corpus		FIRE corpus	
Pronoun	%	Pronoun	%
এ/ e (this)	12.10	এ/ e (this)	14.60
ও/o (he/she)	7.77	ও/o (he/she)	12.57
আমি/ Ami (I)	7.42	এক/ eka (one)	8.19
এক/ eka (one)	7.29	তাঁর/tA.Nra (his)	5.74
তার/tAra (his)	6.04	তার/tAra (his)	5.53
সে/se (he)	5.53	তা/ tA (that)	4.81
তা/ tA (that)	5.01	সে/se (he)	4.79
কোন/ kona (any)	4.69	আমি/ Ami (I)	4.72
তাঁর/tA.Nra (his)	3.46	নিজ/ nija (self)	3.29
যা/ ya (that)	2.68	তিনি/ tini (he)	3.26

Table 2.4 shows the top ten most frequent pronouns in TDIL and FIRE corpus. This ranking is done on the basis of including their inflectional forms. This table also shows that the top ten most frequent pronouns (with their inflectional forms) have the large coverage (i.e. 61.99%) of the pronouns in TDIL corpus and 61.50% of the pronouns in FIRE corpus. From Table 2.4 we note that the personal pronouns are the most frequent ones in the corpus.

2.5 Summary

The first part of this chapter discusses about the existing studies on Bengali pronouns. These studies are mostly in the domain of linguistics and address issues like the pronouns with origin, etymological history, classifications, syntactic and semantic study. This chapter presented a corpus based analysis of Bengali pronouns and explored the morphological structure of Bengali pronouns. Next, various features and exceptions of pronouns are presented with examples and the utility of this analysis is explained for anaphora resolution. The two corpora (TDIL and FIRE) have been exploited for this study. It computes the frequency of pronouns and the coverage in the corpus. The well known LL test is adopted for the significance test. Finally, the relative rank list of pronouns is computed and the result shows that the personal pronouns dominate in the language. Several important aspects are coming out of this chapter which essentially help in configuring a robust anaphora resolution system.

Chapter 3

Basic NLP Tools for Bengali

3.1 Introduction

Before applying the anaphora resolution algorithm the input text is to be processed for annotation to some extent. Annotations like POS tags, NER tags, chunking information are required. In resolution process, other linguistic information related to morphology, person, number, etc. are required. So the input text goes through various pre-processing phases like POS tagging, NER, chunking, morphological analysis, etc. Therefore, the initial challenge is to develop some basic NLP tools for Bengali. For using the machine learning approaches to develop such NLP tools, large amount of annotated data is required to get meaningful results. But as said earlier, Bengali is a resource scarce language and hence the annotated corpus or many basic NLP tools are not yet available. Some cases the resource scarce languages make use of hand crafted annotated data or manually enrich the accuracy of the output of NLP tools [14]. Although some works on developing some basic NLP tools are reported in the literature but most of the systems are either not available in executable mode or difficult to re-produce and hence, we have developed the required basic pre-processing tools as described below.

3.2 POS tagger for Bengali

POS tagging is the process of assigning the appropriate part-of-speech or predefined syntactic categories like Noun, Pronoun, Verb, etc. automatically for each word of a sentence in the text. POS tagging is important for almost all NLP tasks and is considered as the first phase (after text cleaning and tokenization) towards natural language understanding. In POS tagging the words are not considered in isolation, rather in order to find the correct tag of a word, it requires to look at the context in which the word is used in the text.

The earlier approaches for POS tagging were rule based as they were built on linguistic knowledge ([229], [230], [231]). People manually extracted the rules by their linguistic expertise. The advantages of these approaches are that they are able to model the complex and finer information and can handle the exceptional features easily. As a result, the accuracy of the systems were quite high. The disadvantages of this approach are (i) requirement of domain expertise to build all language specific rules and exceptions and (ii) most of the cases the rules and exceptions found in one language are not applicable to other languages. As a result machine learning techniques ([232], [233], [235], [236], [237]) have found their application for POS tagging. These approaches use large POS-annotated corpora to acquire the high-level language features. The main advantage of this approach is that, it is independent of languages; the only constraint is the requirement of large annotated corpus.

Bengali is morphologically rich (Chapter 2). The morphological suffices highly influence the POS tagging of words; the context information, etc. and these features should be considered into account to develop a POS tagger. These features have been incorporated in many machines learning algorithm and we have chosen the approach based on Maximum Entropy principle. Particularly, we have configured the Stanford University POS tagger [62] for Bengali which was originally developed English.

3.2.1 The Maximum Entropy Model

Mathematically the concept of entropy (H) (or uncertainty) is the inversely proportional to the probability (p) (H $\propto$ 1/p) and the entropy is defined by the formula [238]:

$$\mathrm{H} = -\sum p \log p \tag{3.1}$$

The principle of the maximum entropy model is to choose the probability distribution p having the highest entropy out of those distributions subject to the constraints or evidence. The principle is based on two criteria [239]: "*models all that is known*" and "*assume nothing about what is unknown*". The first criteria, "Models all that is known" implies the requirement of satisfying a set of constraints from the given evidence (particularly, given in the training data). And the second one, "assume nothing about what is unknown" implies the requirement of choosing the most uniform (equiprobable) distribution which gives maximum entropy. The principle has been introduced by Berger et al. [239] in NLP domain for POS tagging and later on it has been used in many other NLP tasks ([10], [62], [240]).

The tagger learns a log-linear conditional probability model from the training data, based on the above maximum entropy principle [62]. Given a word and its history (context) h, the model assigns a probability for every tag t from the predefined tag set T of possible tags . The history is usually defined as the sequence of several words in forward and backward direction and their tags. This model is used to estimate the probability of a tag sequence t_1,.......,t_n for a given sentence w_1, w_1,..... w_n as follows.

$$\mathrm{p}(t_1...t_n|w_1...w_n) = \prod_{i=1}^{n} p(t_i|t_1...t_{i-1}, w_1...w_n) \approx \prod_{i=1}^{n} p(t_i|h_i) \tag{3.2}$$

In particular, the constraint is that the expectations of the features for the model are same with the empirical expectations of the features from the training data, i.e.,

$$\mathrm{E}f_j = \widetilde{E}f_j \tag{3.3}$$

where the model's feature expectation is:

$$\mathrm{E}f_j = \sum p(h,t) f_j(h,t) \tag{3.4}$$

and the observed feature expectation is:

$$\widetilde{E}f_j = \sum \widetilde{p}(h_i,t_i) f_j(h_i,t_i) \tag{3.5}$$

where $\widetilde{p}(h_i,t_i)$ denotes the observed probability of (h_i,t_i) in the training data. Thus the constraints force the model to match its feature expectations with those observed in the training corpus. Practically $\mathrm{E}f_j$ can't be computed directly, so the following approximation [240] is used:

$$Ef_j \approx \sum_{i=1}^{n} \widetilde{p}(h_i) p(t_i|h_i) f_j(h_i,t_i) \tag{3.6}$$

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

The feature is defined by the binary valued function which associates a tag with various elements of the context; for example:

$$f_j(h,t) = \begin{cases} 1 & \text{if current word}(w_i) = \textbf{Kolkata} \text{ and } t = \textbf{NNP} \\ 0 & \text{otherwise.} \end{cases} \tag{3.7}$$

Feature selection plays a crucial role in this model. For Bengali the POS tag of a word is highly influenced on its previous word, next word, POS tags of previous and next word, prefix and suffix of the words. Hence we have considered all such features along with their different possible combinations. The history h_i has been defined as follows.

$$h_i = \{w_i, w_{i+1}, w_{i+2}, w_{i-1}, w_{i-2}, t_{i-1}, t_{i-2}, w_{prefix}, w_{suffix}\} \tag{3.8}$$

3.2.2 The Tag set

The first step towards developing POS annotated corpus is to come up with a proper tag set. There are many issues related to POS tag selection. The tag set should capture the fine grained linguistic knowledge (depends on application) and its size (number of tags) is directly proportional to the amount of linguistic information to be captured. If the tag set is too small, the tagging accuracy will be much higher and classification may be easier. But, some important information may be missed because of the coarse grained tag set. On the other hand, if the tag set is too large then it may enrich the supplied information, but the performance of the POS tagger may decrease. So the size should consider the trade-off in-between. The Penn tags are the most popularly used tags for English and it has been used as a benchmark. The IIIT Hyderabad tag set [241] is popular for Indic languages, which is similar to the Penn tag set with some variations. In our experiment twenty five different tags have been selected from the IIIT Hyderabad tag set as shown in Table 3.1.

3.2.3 Training data

The model has been trained with the data available from ICON 2011 [14] and LDC tagged data [75] for Bengali [1]. The ICON 2011 data was originally tagged for anaphora resolution. The tag set that has been used in ICON data is same as described in Table 3.1 but the tag used in LDC Bengali data is different. So, the LDC tags are mapped to the ones in IIIT Hyderabad tag set. The size of the training data is shown in Table 3.2.

[1] https://www.ldc.upenn.edu/

Table 3.1: The tag set for Bengali

Tag	Description	Tag	Description
CC	Conjunction	QO	Ordinal
DEM	Demonstrative pronoun	QF	Quantifier
ECH	Echo words	RB	Adverb
INJ	Interjection	RDP	Reduplicated
INTF	Intensifier	RP	Particle
JJ	Adjective	SYM	Symbol
NEG	Negation	UT	Quotative
NN	Common noun	UNK	Unknown
NNP	Proper noun	VAUX	Verb Auxiliary
NST	Spatial, temporal, ...	VM	Main verb
PRP	Pronoun	WQ	Question word
PSP	Post position	XC	Continuation of compound words
QC	Cardinal		

Table 3.2: The training data size

Source	No. of Sentences	No. of Tokens
ICON	2,018	25,626
LDC	6,229	1,02,941
Total	8,247	1,28,567

3.2.4 Evaluation

The models have been tested on a set of randomly selected news article[1] of 200 sentences (about 2,000 words). And the system performs with an accuracy of 83.7%. It should be noted that about 10% words in the test data were unknown with respect to the training set. We didn't perform the error analysis of the POS tagger in details but we have identified the major errors in the result. The major conflicts were between the following tag pairs: NNP vs NN, JJ vs NN, VM vs VAUX and most of the uncommon words are tagged as NN. Also, we observed that most of the pronouns (PRP), symbols (SYM), numbers (QC) are properly classified.

[1] www.anandabazar.com

3.3 Morphological Analyser

Morphological analyzer is one of the important as well as essential tasks for a number of NLP applications. Most of the Indian languages are highly inflectional and to develop a well suited morphological analyzer is a challenging task. The morphological analyzer returns all the morphemes and their grammatical categories associated with a particular word form. The literature shows that although there are some attempts for morphological analysis of some Indian languages but hardly any computational effort is found for Bengali ([215], [217], [242], [243], [244], [245]).

There are many approaches which are widely used in morphological analyzer, like Corpus Based, Paradigm Based, Finite State Automata (FSA) Based [246], Two-Level Morphology Based [247], Stemmer Based, Suffix Stripping Based [248] etc. Most of the studies in Bengali ([215], [242], [243], [244]) have adopted knowledge-based approaches. These approaches operate by segmenting a word using manually designed rules, which require a lot of linguistic expertise and are also time-consuming to construct. Here we have attempted to develop a finite automaton based morphological analyzer for Bengali. Our analyzer outputs only the root (surface) word, but the system has an ability to produce full-fledged morphological information. The method can be used for any inflected language with minor changes.

3.3.1 The approach

The approach is based on the working principle of finite automaton. The finite automaton takes a string as input, processes the string character-wise with transition, and finally, it reports about a particular pattern of the input string in the form of accept or reject on reaching the final or non-final state respectively. Here we have designed a finite automata to identify the inflectional pattern of Bengali words on reaching the final state. In order to design the automata, the inflectional patterns are collected from a linguistic study of Bengali corpus (TDIL data)[1] and training data (FIRE data)[2] as presented next.

3.3.2 Linguistic Study

Bengali is a highly inflected language and almost all the words are inflected. Based on our corpus study and literature survey, we have observed that the rules of the inflections are mainly based on grammatical, like nominal, pronominal, verbal, animate, inanimate, human, etc. and in some cases they do not follow any rule i.e. irregular basis. For our computational point of view we broadly classify all the words into three categories:
Category 1: This category refers to the class of words where a word (surface) itself is the root, i.e. the word has not been inflected. Examples of such words are: অক্টোবর/ *akTobara*; সফটওয়ার/ *saphaTaoAra*; ফেরত/ *pherata*; নিন্দা/ *nindA*; etc.
Category 2: This category refers to those cases where a root word cannot be extracted from a surface word by formulating any rules. Such words are inflected in an irregular basis. Most of the Bengali pronouns come under this category. Examples of such category are: আমাকে/ *AmAke* (the surface word is আমি/ *Ami*); আমরা/ *AmarA* (the surface word is আমি/ *Ami*); etc.
Category 3: This category refers to those cases where root (surface) word is extracted using a specific rule (i.e. trimming the suffix/prefix). Examples of such category are: সংবাদদাতা/ *saMbAdadAtA* (সংবাদ/ *saMbAda* with suffix দাতা/ *dAtA*); ঘরগুলি/ *gharaguli* (ঘর/ *ghara* with suffix গুলি/ *guli*) etc.

Our approach for morphological analysis emphasized more on in the last category, because most of the words fall under this group. In this category we have identified a SUFFIX LIST of 173 suffixes from TDIL and FIRE data. The sample belonging of the list is { দের/ *dera*, দেরকেই/ *derakei*, টে/ *Te*, এর/ *era*, এরই/ *erai*, এরও/ *erao*, ও/ *o*, কে/ *ke*, কেই/ *kei*, কেও/ *keo*, খানা/ *khAnA*, খানাই/ *khAnAi*, খানি/ *khAni*, গাছা/ *gAChA*, গাছি/ *gAChi*, গুলো/

[1] http://tdil.mit.gov.in/
[2] http://www.isical.ac.in/∼fire/morpho/MET.html

gulo, ...}

3.3.3 The Algorithm

Our approach is a rule based one, implemented by the principle of finite automata. In the above sections we have classified all the words into three categories. For the computational aspect we handled each category in a different manner.

For the Category 1, a DICTIONARY has been built (manually) with the words such that the word itself is the root word. While an input is given, the system first searches the DICTIONARY and if the word is in DICTIONARY then system reports the word itself as the root word otherwise goes ahead to test the next category.

For the Category 2, a MAP (basically a look up table) has been built manually and it looks like a key value pair as shown in the Table 3.3. While an input is given, the system searches the MAP and if found then gives its corresponding map value (i.e. MAP value of আমাকে/*AmAke* is আমি/*Ami*) as the root word otherwise goes to test the next category.

Table 3.3: MAP: Key Value pair

Key	Value (root)
আমাকে/ *AmAke*	আমি/ *Ami*
আমরা/ *AmarA*	আমি/ *Ami*;
ওঁর/ *o.Nra*	ও/ *o*
....	....

For the Category 3, the suffix matching has been implemented by the principle of a Finite Automata (specially a Non-deterministic Finite Automata) using a SUFFIX LIST (mentioned in the previous section).

The architecture of our system is given below.

3.3.4 Construction of the Automata

In our approach for morphological analysis, the most important component is the automata construction. The automata recognize the string character-wise in left-to-right

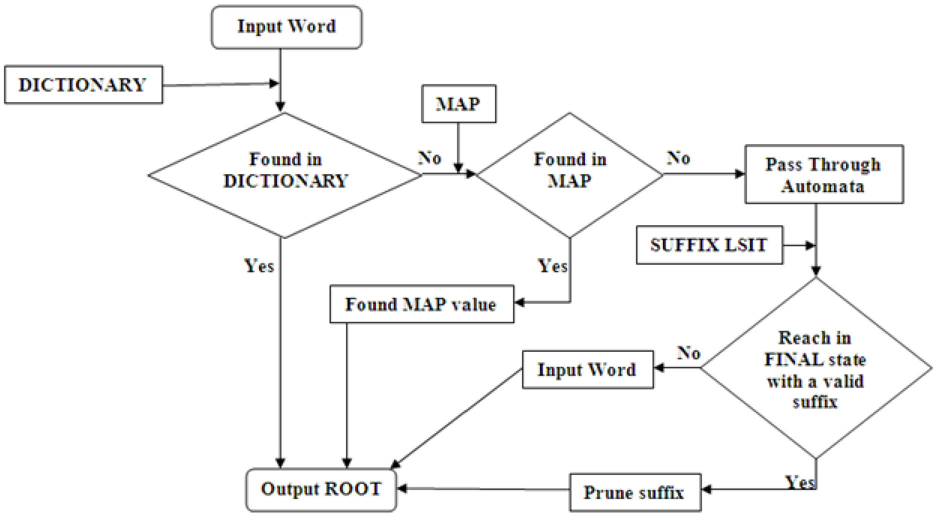


Figure 3.1: Architecture of the System

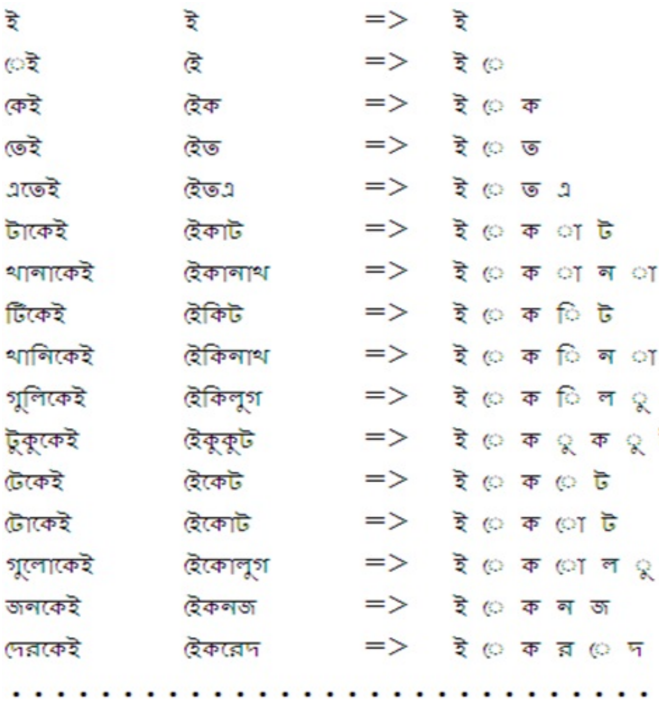
fashion. In Bengali most of the inflections are in suffix position and for suffix matching we can't process the string with left-to-right fashion directly. To tackle this problem, we reverse the string to feed into the automata as described below.

A collection of usual Bengali alphabet along with diacritic forms defines the input alphabet set: { অ, আ, ই, ……, ঔ, ক, খ, গ, ….., া, ি, ী, ু, ূ, ……, ৌ }. A final state is defined when the automata consume a valid suffix. To reduce the computational complexities and for fast processing, the construction has been carried out by the following steps.

Step 1: Reverse all the suffixes, then arrange with respect to the dictionary order and finally, split into characters. The Figure 3.2 shows the representation of all suffices. The first column lists the suffixes; the reverse strings are in the second column and the third column shows the suffixes split into characters.

Step 2: Characters in a suffix is processed in left-to-right fashion and transitions take place inside the corresponding automaton. The Figures (3.3 - 3.6) show transitions corresponding to the suffixes { ই/ *i*, তেই/ *tei*, কেই/ *kei*, ... }.

Step 3: Whenever a suffix completely consumed by the automaton, the state at which this happens is labelled as a final state (The double circle represents the final state shown in figures). The Figures (3.3 - 3.6) show the final states, because { ই/ *i*, তেই/ *tei*, কেই/ *kei*, ... } are the valid suffixes.

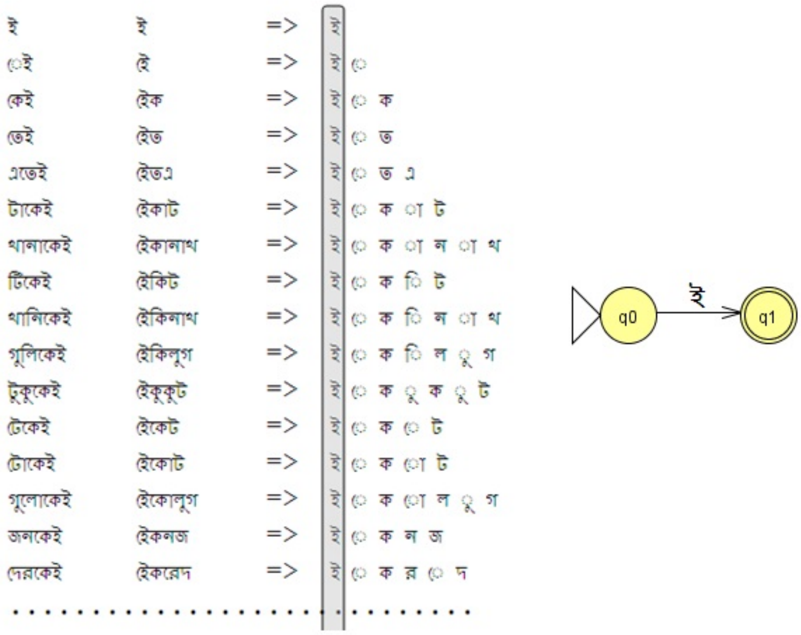


ই	ই	=>	ই
েই	ইে	=>	ই ে
কেই	ইেক	=>	ই ে ক
তেই	ইেত	=>	ই ে ত
এতেই	ইেতএ	=>	ই ে ত এ
টাকেই	ইেকাট	=>	ই ে ক া ট
থানাকেই	ইেকানাথ	=>	ই ে ক া ন া থ
টিকেই	ইেকিট	=>	ই ে ক ি ট
থানিকেই	ইেকিনাথ	=>	ই ে ক ি ন া থ
গুলিকেই	ইেকিলুগ	=>	ই ে ক ি ল ু গ
টুকুকেই	ইেকুকুট	=>	ই ে ক ু ক ু ট
টেকেই	ইেকেট	=>	ই ে ক ে ট
টোকেই	ইেকোট	=>	ই ে ক ো ট
গুলোকেই	ইেকোলুগ	=>	ই ে ক ো ল ু গ
জনকেই	ইেকনজ	=>	ই ে ক ন জ
দেরকেই	ইেকরেদ	=>	ই ে ক র ে দ

. . . .

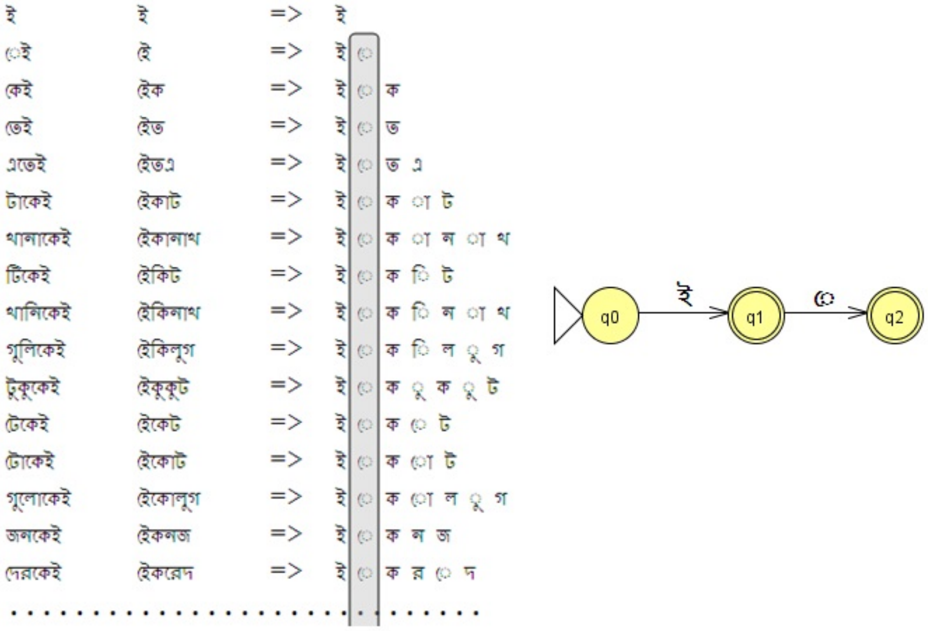
Figure 3.2: Processing of suffixes in Step 1

The construction shows that, an automaton is dedicated with respect to a particular start symbol. For example, the automaton (Figures 3.3 - Figures 3.6) is dedicated to the start symbol ই/*i*. Similarly, for another start symbol we have another automaton. For 173 different suffixes we got thirty different start symbols (ই, ও, ক, গ, ছ, জ, ট, দ, ধ, ন, ত, থ, ম, য, র, ল, ব, ষ, হ, ণ, ড, স, য়, া, ো, ু, ূ, ে, ি, ী) and hence we have thirty different automata. The final construction is done by the combination of all above thirty automata with single one (Figure 3.7) by the λ – transition (the symbol λ represents the nothing or null string). The Figure 3.7 shows the combined automata.



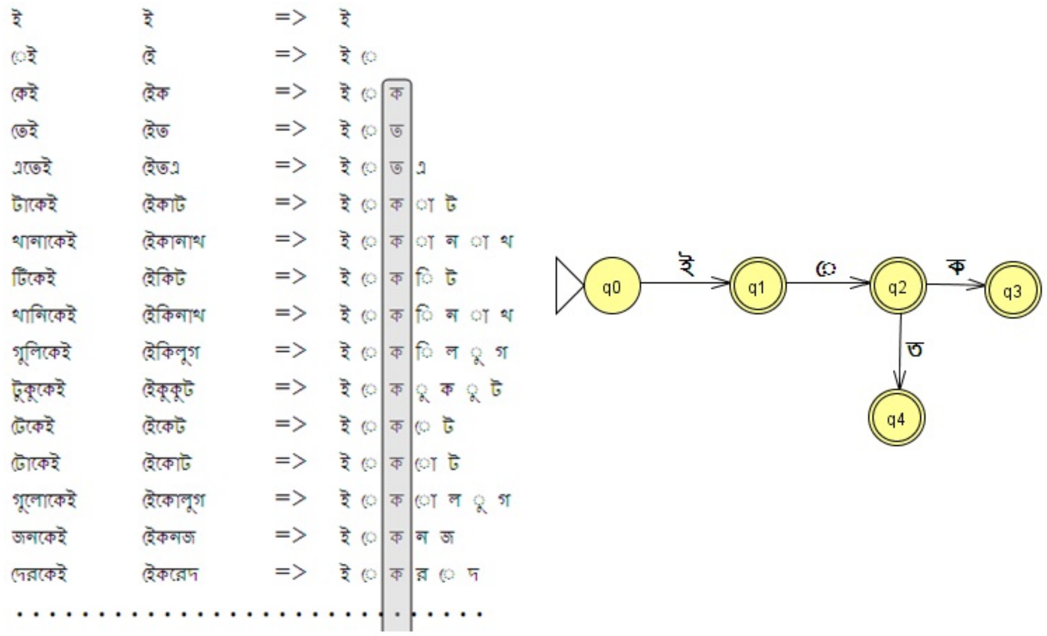
```
ই          ই          =>  ই
েই         ইে         =>  ই ে
কেই        ইেক        =>  ই ে ক
তেই        ইেত        =>  ই ে ত
এতেই       ইেতএ       =>  ই ে ত এ
টাকেই      ইেকাট      =>  ই ে ক া ট
থানাকেই    ইেকানাথ    =>  ই ে ক া ন া থ
টিকেই      ইেকিট      =>  ই ে ক ি ট
থানিকেই    ইেকিনাথ    =>  ই ে ক ি ন া থ
গুলিকেই    ইেকিলুগ    =>  ই ে ক ি ল ু গ
টুকুকেই    ইেকুকুট    =>  ই ে ক ু ক ু ট
টেকেই      ইেকেট      =>  ই ে ক ে ট
টোকেই      ইেকোট      =>  ই ে ক ো ট
গুলোকেই    ইেকোলুগ    =>  ই ে ক ো ল ু গ
জনকেই      ইেকনজ      =>  ই ে ক ন জ
দেরকেই     ইেকরেদ     =>  ই ে ক র ে দ
.............................
```

Figure 3.3: Transition for the character ই/*i*



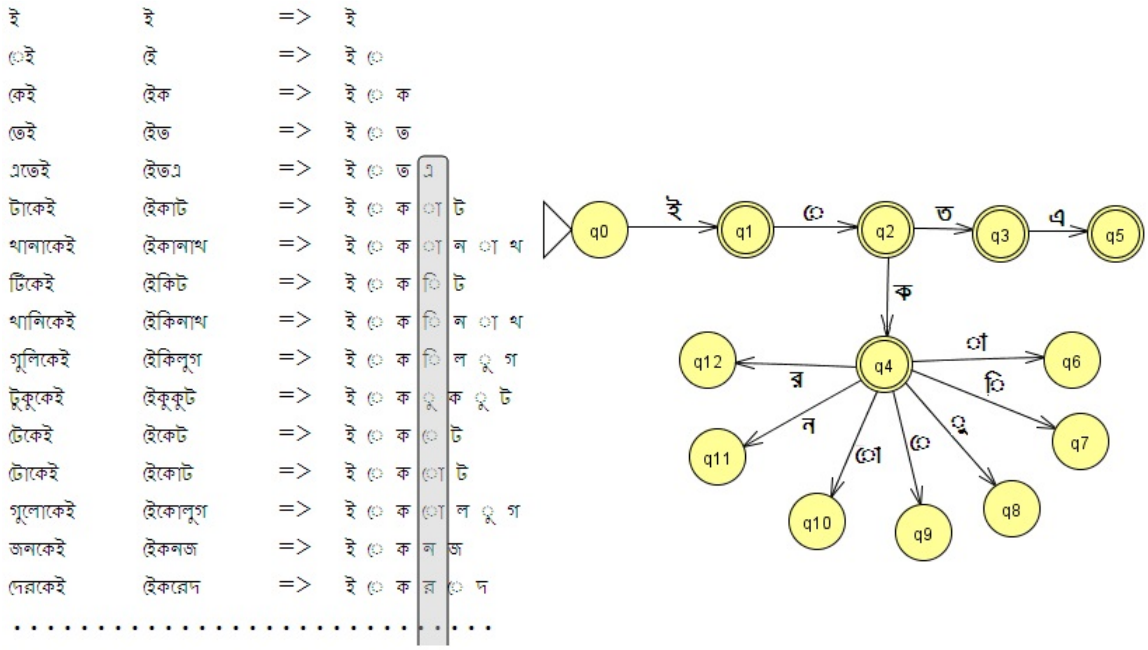
```
ই          ই          =>  ই
েই         ইে         =>  ই ে
কেই        ইেক        =>  ই ে ক
তেই        ইেত        =>  ই ে ত
এতেই       ইেতএ       =>  ই ে ত এ
টাকেই      ইেকাট      =>  ই ে ক া ট
থানাকেই    ইেকানাথ    =>  ই ে ক া ন া থ
টিকেই      ইেকিট      =>  ই ে ক ি ট
থানিকেই    ইেকিনাথ    =>  ই ে ক ি ন া থ
গুলিকেই    ইেকিলুগ    =>  ই ে ক ি ল ু গ
টুকুকেই    ইেকুকুট    =>  ই ে ক ু ক ু ট
টেকেই      ইেকেট      =>  ই ে ক ে ট
টোকেই      ইেকোট      =>  ই ে ক ো ট
গুলোকেই    ইেকোলুগ    =>  ই ে ক ো ল ু গ
জনকেই      ইেকনজ      =>  ই ে ক ন জ
দেরকেই     ইেকরেদ     =>  ই ে ক র ে দ
.............................
```

Figure 3.4: Transition for the characters ই/*i* followed ে/e



```
ই           ই           =>  ই
েই          ইে          =>  ই ে
কেই         ইেক         =>  ই ে ক
তেই         ইেত         =>  ই ে ত
এতেই        ইেতএ        =>  ই ে ত এ
টাকেই       ইেকাট       =>  ই ে ক া ট
থানাকেই     ইেকানাথ     =>  ই ে ক া ন া থ
টিকেই       ইেকিট       =>  ই ে ক ি ট
থানিকেই     ইেকিনাথ     =>  ই ে ক ি ন া থ
গুলিকেই     ইেকিলুগ     =>  ই ে ক ি ল ু গ
টুকুকেই     ইেকুকুট     =>  ই ে ক ু ক ু ট
টেকেই       ইেকেট       =>  ই ে ক ে ট
টোকেই       ইেকোট       =>  ই ে ক ো ট
গুলোকেই     ইেকোলুগ     =>  ই ে ক ো ল ু গ
জনকেই       ইেকনজ       =>  ই ে ক ন জ
দেরকেই      ইেকরেদ      =>  ই ে ক র ে দ
..............................
```

Figure 3.5: Transition for the character ক/*ka* and ত/*ta*



```
ই           ই           =>  ই
েই          ইে          =>  ই ে
কেই         ইেক         =>  ই ে ক
তেই         ইেত         =>  ই ে ত
এতেই        ইেতএ        =>  ই ে ত এ
টাকেই       ইেকাট       =>  ই ে ক া ট
থানাকেই     ইেকানাথ     =>  ই ে ক া ন া থ
টিকেই       ইেকিট       =>  ই ে ক ি ট
থানিকেই     ইেকিনাথ     =>  ই ে ক ি ন া থ
গুলিকেই     ইেকিলুগ     =>  ই ে ক ি ল ু গ
টুকুকেই     ইেকুকুট     =>  ই ে ক ু ক ু ট
টেকেই       ইেকেট       =>  ই ে ক ে ট
টোকেই       ইেকোট       =>  ই ে ক ো ট
গুলোকেই     ইেকোলুগ     =>  ই ে ক ো ল ু গ
জনকেই       ইেকনজ       =>  ই ে ক ন জ
দেরকেই      ইেকরেদ      =>  ই ে ক র ে দ
..............................
```

Figure 3.6: Transition for the character এ/*e*, র/*ra*, ...

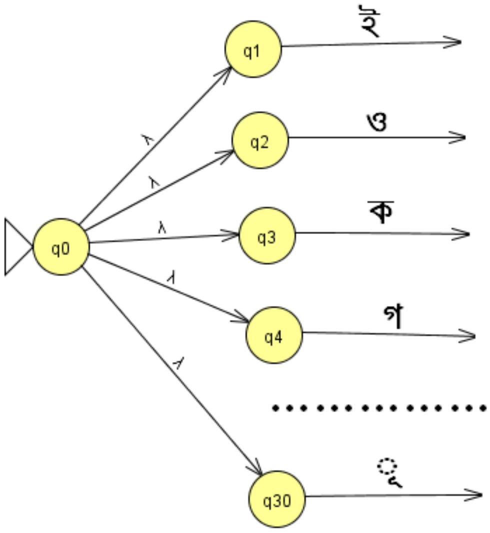


Figure 3.7: Combined automata

3.3.5 Decision in Final State

When a final state is reached we know that a valid suffix has been encountered. Next task is to strip the suffix to get the root word. But for the complex morphology of Bengali we can't always handle this task of suffix stripping in such a simple way. Some complicated cases have been considered below.

Case 1: when the input string is সবগুলোরই/ *sabagulorai* (actually automata takes input in reverse order of input string, i.e. as ইরোলুগবস => ই র ো ল ু গ ব স). In this case the system will identify the three valid suffixes (since there are three final states) namely ই/*i*, রই/ *rai*, and গুলোরই/ *gulorai* (shown in Figure 3.8).

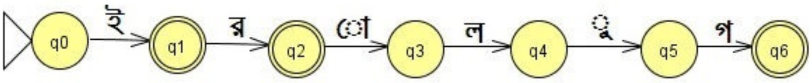


Figure 3.8: Processing of 'সবগুলোরই'/*sabagulorai*

In this case our system will consider the three possible root words সবগুলোর/ *sabagulora*, সবগুলো/ *sabagulo* and সব/ *saba*. The system has the capability (user configurable) to produce all the root words and by default gives the root corresponding to the longest suffix i.e. সব/ *saba*.

Case 2: when input string is শিক্ষীতা/ *shikShItA*, in this case our system finds the suffix as তা/ *tA* and root word is শিক্ষী/ *shikShI*, which is not correct. In such cases, some heuristics are used to output শিক্ষা/ *shikShA* as the valid root instead of শিক্ষী/ *shikShI*.

3.3.6 Evaluation

The system has been evaluated using the FIRE-MET [1] test data. The evaluation was done by the FIRE 2012 organizers (participants have asked to submit their systems). Test data comprised of 30,000 surface words in Bengali. The results were evaluated manually and Mean Average Precision [249] was considered as the evaluation metric. The Mean Average Precision score of our system was reported as 0.3159 where score of the baseline result provided by FIRE-MET was 0.2740.

3.4 Named Entity Recognition (NER)

The term “Named Entity” refers to proper nouns appearing in a discourse. Named Entity Recognition (NER) is the task of identifying a proper name as name of person, organization, or location, or whether it is a numeric expressions referring time, date, or money, etc. The NER plays significant role in many NLP applications, namely in machine translation, question-answering, information retrieval, anaphora resolution, etc. Unlike in English, the development of an NER system is difficult for most of the Indic languages due to some inherent problems like absence of capitalization, multiple meaning of a word, dictionary words are used as names, etc. Though there is some research on developing Bengali NER systems ([250], [251], [252], [253]), but no off-the-shelf system is available in the public domain. Here we have described a rule based approach as for designing an NER for Bengali.

[1] http://www.isical.ac.in/∼fire/morpho/MET.html

3.4.1 Challenges in designing Bengali NER

There are so many issues in using named entities in Bengali ([250], [252]). Based on their importance with respect to NER some of these are stated below.

Capitalization: Capitalization information (the start letter is capitalized) plays a very important role in identifying name entities. Though this feature is available in English and most of the European languages, but this feature is absent in Bengali.

Example: নিউটন/ *niuTana* (Newton), কলকাতা/ *kalakAtA* (Kolkata), etc.

Use of dictionary words: Indian person names are more diverse compared to the other languages and a lot of these words can be found in the dictionary with some other specific meanings.

Example: কবিতা/ *kabitA* may be a name of a person or mean *poem* in Bengali; similarly পদ্ম/*padma* may be a name of a person or mean *lotus* in Bengali.

Morphologically rich: Bengali is a highly inflected language providing one of the richest and most challenging sets of linguistic and statistical features resulting in long and complex word forms.

Example: কটক/ *kaTaka* (Cuttack) is a proper noun i.e. the name of a location, whereas কটকি/ *kaTaki* appears as an adjective in কটকি জুতো/ *kaTaki juto* (shoes made in Cuttack) to imply a special kind of shoes.

Sentence structure: Bengali is a relatively free-order language. Thus, named entities can appear in any position of the sentence. For example:

- আমি কলকাতা যাই/ *Ami kalakAtA yAi* (I go to Kolkata)
- কলকাতা যাই আমি/ *kalakAtA yAi Ami* (I go to Kolkata)
- আমি যাই কলকাতা/ *Ami yAi kalakAtA* (I go to Kolkata)

All the sentences are valid implying the same meaning.

Spelling variations: In Bengali proper noun, especially in case of the name of a person or a location, frequent variation of spelling is observed.

Example: The name of a person অপূর্ব/ *apUrba* (Apurba), অপুর্ব/ *apurba* (Apurba), অপূর্ব্ব/ *apUrbba* (Apurba) both are valid. Similarly, the city name Kolkata is written in different forms like কলকাতা/ *kalakAtA*, কোলকাতা/ *kolakAtA*, কলিকাতা/ *kalikAtA*, etc.

Resource scarce: Bengali is a resource scarce language. The basic NLP tools (annotated corpora, name dictionaries, good morphological analyzers, POS taggers, chunkier, etc.) are either not available or even if available, the tools do not provide satisfactory accuracy.

Web source: Web sources for name lists are available in English, but such lists are not available for Bengali forcing the use of transliteration for creating such lists.

3.4.2 Our Approach

We have followed a rule-based approach for NER. The rules used by the system are mined from the training data (NER training data from ICON 2013[1] and FIRE 2013[2]) and literatures (NER for South and South East Asian Languages[3]). In Bengali, most of the named entities (e.g. Person, Date, Time, Year, Quantity, etc.) have some context either before or after the word. For example, in case of a person, most of the cases it appears with some honorific addressing terms (ডঃ/ *DaH*(Dr.), মিঃ/ *miH*(Mr.), মিসেস/ *misesa*(Mrs.), শ্রী/ *shrI*(Sri), etc.) or followed by middle name (চন্দ্র/ *chandra*, বরন/ *barana*, লাল/ *lAla*, নাথ/ *nAtha*, etc.) or surname (ঘোষ/ *ghoSha*, বসু/ *basu*, রায়/ *rAy*, মুখার্জী/*mukhArjI*, etc.). Similarly, in case of stating a distance it must have some numeric value (in numerical form or in word in Bengali script) followed by with some distance measuring unit (মাইল/ *mAila* (mile), ইন্চি/ *inchi* (inch), ফুট/ *phuTa* (feet), কিমি/ *kimi* (km), মিটার/ *miTAra* (meters), etc.). The unit may appear before (e.g., মাইল তিনেক/ *mAila tineka* (three miles)) or after (তিন মাইল/ *tina mAila* (three miles)) the numeric value. Such contexts available for almost all the types of named entities in Bengali. Based on our observation for most of the named

[1] http://ltrc.iiit.ac.in/icon/2013/nlptools/
[2] www.isical.ac.in/∼fire/2013/
[3] http://ltrc.iiit.ac.in/ner-ssea-08/index.cgi

entities, such context related words are limited (except the person and location type). This fact is one of the significant clues to develop an NER system in Bengali. In our experiment, the tag set is initially classified as of two types: the named entity **without numerical expression** and named entity **with a numerical expression**. The first category is further subdivided as Person, Location, Organization and Facilities. The second category is also subdivided as, Distance, Money, Quantity, Count, Time, Year, Month, Date and Period. We have defined a context dictionary containing all such category specific context words for Bengali to recognize named entities.

The dictionary contains the context words of each named entity, i.e. person, location, etc.. The system makes use of this dictionary to identify the name entity type for a given context. The dictionary is built from the training data for NER provided by FIRE 2013 and ICON 2013. The following contexts are considered to build the dictionary.

Honorific context: In Bengali, a set of honorific addressing terms used to esteem the honour of a person. These terms are generally used either before (ডঃ/ *DaH* (Dr.), মিঃ/*miH* (Mr.), মিসেস/ *misesa* (Mrs.), শ্রী/ *shrI* (Sri), etc.) or after the name (বাবু/ *bAbu*, মশাই/ *mashAi*, সাহেব/ *sAheba*, etc.). In our system we have used 45 such terms found from training data.

Surname context: Surname always comes after the name of a person and hence collection of common surnames (ঘোষ/ *ghoSha*, বসু/ *basu*, রায়/ *rAy*, মুখার্জী/ *mukhArjI*, etc.) helps to resolve the entity's name.

Middle name context: Like surnames a list of predefined common middle name (চন্দ্র/*chandra*, বরন/ *barana*, লাল/ *lAla*, নাথ/ *nAtha*, etc.) also helps to resolve some named entities.

Relational term context: In Bengali community some relational terms (দা/dA (elder brother), দাদা/ *dAdA* (elder brother), দি/ *di* (elder sister), দিদি/ *didi* (elder sister), কাকা/ *kAkA* (uncle), কাকু/ *kAku* (uncle), etc.) are used together (immediate next) with name. Example: বাদল-দা/ *bAdala-dA*, মিঠুদি/*miThudi*, মিঠু দি/*miThu di*, etc.

Abbreviated form of person names: A dot (.) separated abbreviated form (কে. ডি. সিংহ/ *K. D. siMha*) with an honorific context before or after also represents the person's name.

Context for location: Some suffixes (গ্রাম/ *grAma*, পুর/ *pura* , নগর / *nagara*, গঞ্জ/ *ga∼nja*, etc.) are location related context terms. Presence of such suffixes is checked to identify location names, i.e., name of some places. Presence of some suffixes (or next word) (দ্বীপ/ *dbIpa*, দ্বীপপুঞ্জ/ *dbIpapuj∼na*, সৈকত/ *saikata*, বিচ/ *bicha*, etc.) identifies location with sub category, i.e., landscape. Some suffixes (মন্দির/*mandira*, মল/*mala*, বাজার/ *bAjAra*, etc.) (or next word) preceded by a name can be used to identify an NE with sub category ”man-made”.

Distance context: The numerical value, followed by a distance measuring unit (মাইল/ *mAila* (mile), ইন্চি/ *inchi* (inch), ফুট/ *phuTa* (feet), কিমি/ *kimi* (km), মিটার/ *miTAra* (meters), etc.) or a distance measuring unit followed by a numerical value identifies the category related to NE, distance.

Quantity context: The numerical value, followed by a quantity measuring unit (কিলো/ *kilo* (kilogram), কেজি/ *keji* (kilogram), লিটার/ *liTAra* (liter), ডিগ্রী/ *DigrI* (degree), %, etc.) or a quantity measuring unit followed by a numerical value helps in identify quantity category.

Month context: Names of months used in Bengali (জানুয়ারী/ *jAnuArI* (January), জানু / *JAnu* (Janu), ফেব্রুয়ারী/ *phebruArI* (February), ফেব্রু/ *phebru* (Feb), etc. বৈশাখ/ *baishAkha*, জ্যৈষ্ঠ/ *jyaiShTha*, etc.) and month related terms (মাস/*mAsa*, মাসে/ *mAse*, etc.) identify time related NEs.

Year context: A numerical value, followed by a year related term (সাল/ *sAla* (year), শতাব্দি/ *shatAbdi* (century), খ্রীষ্টাব্দ/ *khrIShTAbda* (Christian era), খ্রীষ্ট-পূর্ব/ *khrIShTa-pUrba* (BC), etc.) or year related term followed by a numerical value also identifies time related NEs.

Period context: A numerical value, followed by a period related term (বছর/ *baChara* (year), মাস/ *mAsa* (month), দিন/ *dina* (day), যুগ/ *yuga* (era), আমল/ *Amala* (period), সপ্তাহ/ *saptAha* (week), etc.) or a period related term followed by a numerical value indicates presence of a time related NE.

Time context: A numerical value, followed by a time measuring unit (সেকেন্ড/ *sekenDa* (second), ঘন্টা/ *ghanTA* (hour), দিন/ *dina* (day), সকাল/ *sakAla* (morning), বিকাল/ *bikAla* (afternoon), etc.) or some time measuring unit followed by a numerical value identifies a time related NE.

Date context: A numerical value, followed by any month name (জানুয়ারী/ *jAnuArI* (January), জানু / *JAnu* (Janu), ফেব্রুয়ারী/*phebruArI* (February), ফেব্রু/*phebru* (Feb), etc. বৈশাখ/ *baishAkha*, জ্যৈষ্ঠ/ *jAnuArI*, etc.) or date related term (সোমবার/ *somabAra* (Monday), মঙ্গলবার/ *ma∼NgalabAra* (Tuesday), etc.) indicates the presence of a date related NE.

Money context: A numerical value, followed by a money measuring unit (টাকা/*TAkA* (rupee), হাজার/ *hAjAra* (thousand), ডলার/ *DalAra* (dollar), লাখ/*lAkha* (lakh), Rs., USD, etc.) or a money measuring unit followed by a numerical value represents quantity related NE category. Note that a numeric value may have Bengali digits (০, ১, ২, ৩, ৪, ৫, etc.), English digits (0, 1, 2, 3, 4, 5, etc.) and can be written in Bengali words (এক/*eka* (one), দুই/*dui* (two), তিন/*tina* (three), etc.) or by using other variations (তিনেক/*tineka*, চারেক/*chAreka*, পাঁচেক/*pA.Ncheka*, etc.).

Count context: A numerical value, followed by a count related term (টি/*Ti*, টা/*TA*, জন/*jana*, জোড়া/*jorA* (pair), etc.) or a count related term followed by a numerical value represents some count. Example: ৩১টি/*31Ti* (31), তিনজোড়া/*tinajoDaA* (three pairs), etc.

Entertainment context: The entertainment context may have words like উৎসব/*utasaba* (festival), সাফারি/*sAphAri* (trip), পেরাগ্লাইডিং/*perAglAiDiM* (paragliding), etc.

Organization context: The organization related context can have words like করপোরেশন/ *karaporeshana* (corporation), সংস্থা/ *saMsthA* (agency), লিমিটিড/ *limiTiDa* (limited), etc. to identify the organization related NEs. A dot(.) separated abbreviated form (e.g., C.M.C) also represents the organization.

Facility context: The facility context may have words like বিশ্ববিদ্যালয়/ *bishbabidyAlay* (university), কলেজ/ *kaleja* (college), রিসর্ট/ *risarTa* (resort) etc. to indicate the facility related NEs.

Skip word: The reduplicate words (repetition of same word, রাম-রাম/*rAma-rAma*, হরি-হরি/*hari-hari*), words with a classifier (টি/*ti*, গুলি/*guli*, খানি/*khAni*, টাকে/*tAke*, etc.), echo words [250] (গাছে-টাছে/*gAChe-TAChe*, খাবার-দাবার/*khAbAra-dAbAra*, etc.) etc. are not named entities. We maintain a list of reduplicated and echo words obtained from the training data. This feature has been used in the system designed by Chaudhuri et. al. [250]. In our system, we also have included all the pronouns and conjunctions as skip words.

Our NER system is based on the context information discussed above. From the context information we can broadly categorize named entities into two types, i.e. NE's with the numeric expression and NE's without numeric expression. The name entities with numerical values are generally one from the following list: {Distance, Money, Quantity, Count, Time, Year, Month, Date, and Period} and the NE's without numerical values are generally one from {Person, Location, Organization and Facilities } etc. The core of our NER algorithm is based on context matching as described below:

Step 1: If a word is a skip word, then skip otherwise go to Step 2.

Step 2: For an input word find its context. The context of a word (w) is defined as the one word previous and one word after; or two previous and two next words (denoted by ±1 context and ±2 context).

Step 3: If words having numeric value, then go to Step 4 else go to Step 5.

Step 4: With the context information around the word (±1 context & ±2 context) search the context dictionaries (corresponding to distance, quantity, month, etc.). If a match found, then resolve otherwise skip.

Note that, the context is defined in two way, i.e. the context (±1) is tried first and then the context (±2) is applied. Now if context ±1 matches, but ±2 context does not match then resolve with respective name entities in the first context. But when ±1 does not match or both the contexts have matches, then resolve with respective name entities using the context (±2).

Step 5: If the word is a dot (.) separated abbreviated name, then take its context and search the context dictionaries corresponding to "organization" else search the context dictionaries (corresponding to a person, location, entertainment, etc.). If a match found, then resolve otherwise skip.

The illustration of the algorithm with examples is given below.

- **... দীর্ঘ ৪৫০ বছরের সাতবাহন রাজত্বে ...**/*dIrgha 450 baCharera sAtabAhana rAjatbe* (... during the 450 years of Satavahanas kingdom...)

In this example, the token ৪৫০/*450* is identified as a word with numeric value. Its context ±1, gives two words, i.e., দীর্ঘ/*dIrgha* and বছরের/*baCharera* and the context ±2 gives one more word, i.e., সাতবাহন/*sAtabAhana*. When these context words are searched in the dictionaries, for the context ±1 a match is found with periods (because of the word, বছরের/*baCharera*) and hence the expression, ৪৫০ বছরের/*450 baCharera* is resolved as "period" type NE's.

- **... সাড়ে পাঁচ হাজার বছর পূর্বে ....**/*sADe pA.Ncha hAjAra baChara pUrbe* (... five and a half thousand years ago ...)

In this example, the token পাঁচ/*pA.Ncha* (five) has numeric representation. Its context ±1 contains words সাড়ে/*sADe* and হাজার/*hAjAra*. When this context is searched in the dictionaries, it matched with the count context because of the word হাজার/*hAjAra*. The context ±2 also matches with period context because of the word বছর/*baChara* (year). Finally, the match for second context is chosen and the expression সাড়ে পাঁচ হাজার বছর/*sADe pA.Ncha hAjAra baChara* (five and a half thousand years) is classified as "period" type NE.

- **...** এই আইল কে টেংগু সাহেবের আইল বলে উল্লেখ করতো **...**/*ei Aila ke TeMgu sAhebera Aila bale ullekha karato* (... this ridge used to be referred as the ridge of Mr. Tengue)

In this example, while classifying the token টেংগু/*TeMgu* its context ±1 gives two words, কে/*ke* and সাহেবের/*sAhebera* and context ±2 gives আইল/*Aila* and আইল/*Aila*. In this example, only context ±1 match in the dictionary with the honorific context because of the word সাহেবের/*sAhebera*. Hence the NE টেংগু সাহেবের/*TeMgu sAhebera* (Mr. Tengue) is classified as "person".

3.4.3 Evaluation

The initial version of the system has been evaluated by the FIRE 2013 organization. The evaluation metrics used are Precision, Recall and F-measure. The FIRE2013 organizers have reported the following result[1] for our system: Precision = 23.69; Recall = 28.02 and F-measure = 25.68. Later on, we have revised our system by extending the context dictionaries and evaluation of the modified system with the data as mentioned earlier in section 3.2.4 gave the following result, Precision = 28.53; Recall = 36.20 and F-measure = 31.91.

[1] http://au-kbc.org/nlp/NER-FIRE2013/index.html

3.5 Summary

This chapter attempts to overcome the basic resource scarcity for the anaphora resolution. In order to annotate data with at least POS tags, NE tags, chunking information and some morphological information, this chapter presents our effort for developing some basic NLP tools like POS tagger, NER and morphological analyzer. The Stanford University POS tagger [62] has been retrained for Bengali language. The tagger originally was developed for English language based on Maximum Entropy principal and we have trained it for Bengali language. To train the system, annotated data is collected from various sources ([14], [75], and [241]). The POS tagger achieved about 84% accuracy. Next, the chapter concentrates on developing a Morphological analyser. Our approach is a rule based one, implemented by the principle of finite automata. The automata is used for suffix matching against a predefined list of suffixes. The suffix list is collected from the training data of FIRE-MET test data. The inflections of a large number of Bengali words have irregular basis and these are encountered by using a look up table. Finally, the chapter develops a rule based NER and it is done with untagged data. The rules are formulated with the context information of a named entity. The honorific context, surname context, middle name context, relational term context, context for location, distance context, etc. are used. Many of such tools are now made available online[1] in order to facilitate future NLP research in Bengali.

[1] http://www.isical.ac.in/∼utpal/resources.php

Chapter 4

GuiTAR-based Bangla PAR

4.1 Introduction

From the last decades, there is a growing interest in using anaphora resolution module as a component of the processing pipeline for several NLP applications such as information extraction [1], question answering ([2], [3]), text summarisation [4], etc. GuiTAR is a General Tool for Anaphora Resolution [5]. It works as an off-the-shelf component of a language processing pipeline. The system has been designed following a modular approach with flexible architecture [6] which supports varieties of input format. For anaphora resolution system several input formats (XML [5], CoNLL [7], text [8], etc.) are in use based on corpora and underlying technology like part-of-speech tagger, chunker, parsers, etc. And hence a support for different input formats is extremely important for a flexible architecture. Actually the GuiTAR system was designed for the English language and it resolves pronouns using the Mitkov et. al. algorithm [8]. We have successfully configured the system for Bengali language and evaluated the system with a publicly available data set for Bengali.

4.2 GuiTAR: A General Tool for Anaphora Resolution

The system GuiTAR has been made to achieve a modular design and independent of input format, so that different resolution algorithms for different types of anaphoric expressions can be tested, as well as algorithms that deal with all types of anaphoric expressions can be incorporated. From a process-oriented perspective GuiTAR can be viewed as consisting of two parts: 1) construction of a discourse model and 2) resolution of anaphora.

4.2.1 Architectural Design

The architecture of the system is shown in following Figure 4.1 (taken from Poesio et. al. [5]). It takes as an input either an XML or a raw text and produces an XML file annotated with anaphoric links, and an evaluation of AR performance with reference to an annotated corpus. The XML is also used as the data interchange format between modules. Next, the main parts of the pipeline, the pre-processing module, the MAS-XML format, etc. are explained in more details.

4.2.2 Pre-processing

The Anaphora Resolution Module is designed to take input syntactic information in an XML format, called MAS-XML (MAS stands for Minimum Anaphoric Syntax). The aim of the MAS-XML is to include the minimal information required by AR systems and thus the task of pre-processing module is to translate a given input format to a MAS-XML compliant format. There are many way to convert an input into MAS-XML i.e. text-to-MAS-XML on the LT-XML suite of tools [9] or text-to-MAS-XML with Charniak's full parser [10] or any XML-to-MAS-XML converters.

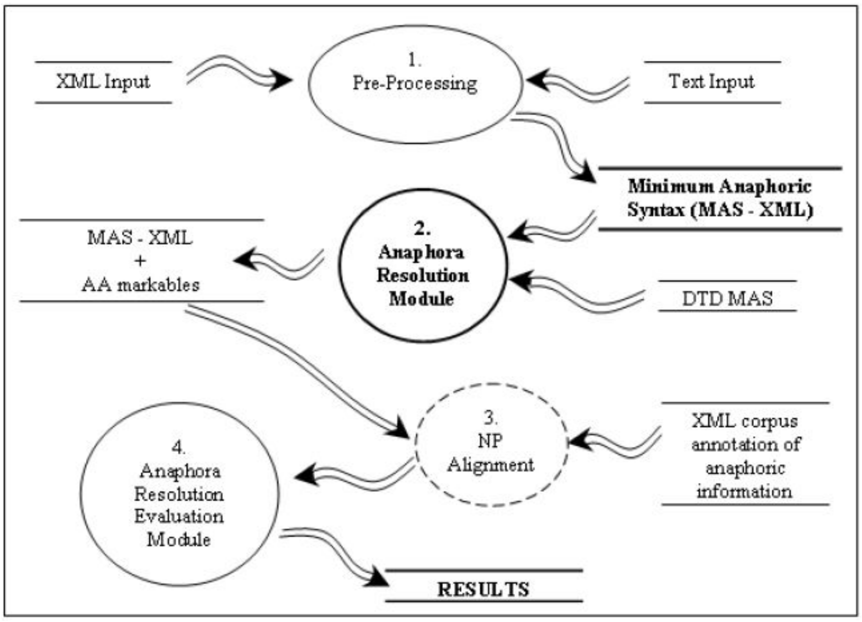

XML Input
1. Pre-Processing
Text Input
Minimum Anaphoric Syntax (MAS - XML)
MAS - XML + AA markables
2. Anaphora Resolution Module
DTD MAS
3. NP Alignment
XML corpus annotation of anaphoric information
4. Anaphora Resolution Evaluation Module
RESULTS

Figure 4.1: Data flow model of the processing pipeline in GuiTAR

4.2.3 MAS-XML (Minimum Anaphoric Syntax XML)

The system GuiTAR uses the XML format for message passing, i.e. for input, output and for pre-processing. The format is in MAS-XML (Minimum Anaphoric Syntax - XML), based on the GNOME mark-up scheme [11], that contains the minimum necessary information required by anaphoric resolvers. The required information can be easily produced from the output of a full parser and possibly (relatively) easily approximated starting from the output of POS tagger or chunker.

Nominal expressions (NE) (mean chunks of words forming simple noun groups) are the main processing units of an anaphora resolution system. An NE is marked with an XML tag (e.g., **<NE>**) and uniquely identifiable corresponding to an **ID** attribute with a unique value. Each NE is also categorized based on syntactic feature and it is defined by the attribute **CAT** (categorized) in the XML. Similarly the other agreement features (person, number and gender) are defined by the attributes **PER** (person), **NUM** (number) and **GEN** (gender), respectively (systems that cannot extract such features could either include ‘guessers’ in their pre-processors or should supply under specified values). As part of the actual structure of an NP, the modifiers and heads are marked as **<MOD>** and **<NPHEAD>** tags respectively. Finally, MAS-XML also requires tokens to be marked with a tag **<W>** with their POS attributes **P**, and sentences are marked with an **<s>** tag.

The following tagged format shows how the NE “the fragile eggs” would be marked-up in MAS-XML:

```
............................
<NE ID=“ne139” CAT=“the-np” PER=“per3” NUM=“plur” GEN=“neut”>
        <W ... P=“DT”>The</W>
        <MOD ID=“m89” type=“preadj”>
                <W ... P=“JJ”>fragile</W>
        </MOD>
        <NPHEAD>
                <W P=“NNS” C=“ ”>eggs</W>
        </NPHEAD>
</NE>
............................
```

The system output (resolution information) is expressed by an additional tag **<ANTE>**. The antecedent (i.e. **<ANTE>**) element has attributes CURRENT specifies the anaphoric expression and REL specifying a semantic relation; it contains one or more **<ANCHOR>** elements specifying the antecedent. The following tagged format shows the antecedent tag:

```
............................
<ANTE CURRENT=“ne139” REL=“ident”>
                <ANCHOR ANTECEDENT=“ne112” />
```

</ANTE> ....

4.2.4 Anaphora Resolution Algorithm in GuiTAR

The GuiTAR[1] (version 3.0) resolves four types of anaphora: Personal Pronouns, Possessive Pronouns, Definite Descriptions and Proper Nouns. In our corpus based study in Chapter 3 we have seen that personal pronouns are the most frequently occurring pronouns and hence they cover the major part of the pronouns.

Personal Pronouns: to resolve this category of pronouns Mitkov Anaphora Resolution Algorithm (MARS) [8] has been used with slight modifications to exploit the discourse model.

Possessive Pronouns: here also MARS have been exploited to resolve the pronouns. But in this case of binding information a binding constraint is used that precludes co-referential links between the possessor and the possessed cases (e.g., *his* and *friend* in *his friend*).

Definite Descriptions: to resolve definite description the 'direct anaphora' resolution algorithm by Poesio et. al. [12] is used.

Proper Nouns: to resolve proper noun the algorithm proposed by Bontcheva et. al. [13] has been used.

4.3 Customization of GuiTAR for Bengali

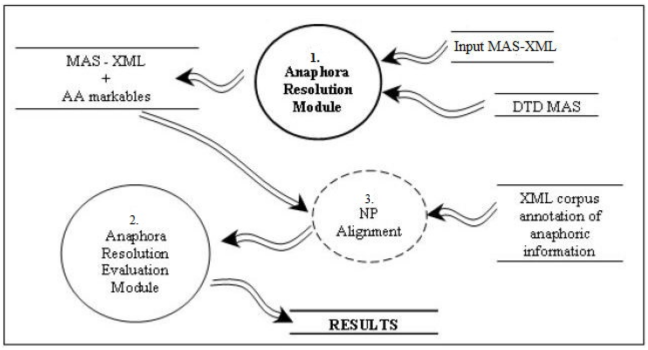
As described above the architecture is quite flexible and it is easily configurable for other languages and the system covers personal pronoun i.e. the most frequently occurring pronouns in Bengali. To adopt GuiTAR for other languages the main concern is to identify the dependencies and modify the system accordingly. In this case the dependencies can be broadly categorized as: dependency in pre-processing (to build MAS-XML) phase and dependency in algorithm (language related issues). These two dependencies

[1] http://cswww.essex.ac.uk/Research/nle/GuiTAR/

are described below.

4.3.1 Dependency in Preprocessing

As shown in the above architecture (Figure 4.1) the pre-processing phase builds MAS-XML from raw input text or other XML input. The lots of underline work (like POS tagging, NER, Chunking, Parsing) is involved in this phase. But as mentioned earlier, all such sophisticated tools are not available for Bengali language and hence it is not possible to build the MAS-XML automatically from the raw text. Here we have built the MAS-XML externally from the data with necessary information (POS tag, NE tag, chunking information, NP tag) with the same format as discussed in section 4.2.3 and supplied to the algorithm directly. The data set [14] used by us is in column format (CoNLL [7]) and all pre-processing information (POS tagged, NE tagged, Chunking) is annotated there. Next, this column format is converted to MAS-XML format which is fed to the system. As an example, the data in Table 4.1 is converted into the required MAS-XML format as shown in Table 4.2. Hence, the modified architecture for configured GuiTAR is shown in Figure 4.2. The detailed description of the data format is shown in Table 4.3.



MAS - XML + AA markables
1. Anaphora Resolution Module
Input MAS-XML
DTD MAS
3. NP Alignment
XML corpus annotation of anaphoric information
2. Anaphora Resolution Evaluation Module
RESULTS

Figure 4.2: Data flow model of the reconfigured GuiTAR

Table 4.1: Sample Column Format Data

......							
story2.txt	0	0	সবশেষে	NN	B-NN	o	-
story2.txt	0	1	তার	PRP	B-NP	(22)	-
story2.txt	0	2	মনে	NN	B-NP	o	-
story2.txt	0	3	হলো	VM	B-VGF	o	-
story2.txt	0	4	এদিকে	NN	B-NP	o	-
story2.txt	0	5	আর	QF	B-NP	o	-
story2.txt	0	6	কোথাও	PRP	I-NP	o	-
story2.txt	0	7	বাঘ	NN	B-NP	B-LIVTHINGS	-
story2.txt	0	8	নেই	VM	B-VGF	o	-
story2.txt	0	9	।	SYM	I-VGF	o	-
......							

Table 4.2: Converted MAS-XML data

```
............................
    <S ID = ID="s81">
        <NE ID="ne237">
            <W P="NN">সবশেষে</W>
        </NE>
        <NE ID="ne238" HON="neu" CAT="pers-pro" PER="per3" NUM="sing">
            <NPHEAD ID="h54">
                <W P="PRP">তার</W>
            </NPHEAD>
        <NE ID="ne239">
            <W P="NN">মনে</W>
        </NE>
    ............................
```

Table 4.3: Description of Data

Column	Type	Description
1	Document Id	Contains the file name
2	Part number	File are divided into part numbered
3	Word number	Word index in the sentence
4	Word	Word itself
5	POS	POS of the word
6	Chunking	Chunking information using IOB format
7	NE tags	Name Entity Information is given
8	Co-reference	Co-reference information

4.3.2 Dependency in Algorithm

GuiTAR resolves personal and possessive pronouns using an implementation of the MARS pronoun resolution algorithm [8]. The algorithm uses several syntactic agreements (based on person, number and gender) but the gender agreement is not applicable in Bengali. Because the gender agreement has no role in Bengali but the honorific agreement has an important role in a pronoun antecedent relationship (Chapter 2). Hence, we have removed the gender agreement and incorporated the honorific agreement following the work in [15]. Moreover, the way pronouns are divided in MARS implementation is not always relevant in Bengali. For example, we do not differentiate between personal and possessive pronouns, but they are separately treated in MARS. In our case, we have only considered the personal and reflexive pronouns while applying MARS based implementation for anaphora resolution.

Also, in case of more than one antecedent found, GuiTAR resolves it by using five antecedent indicators, namely aggregate score, immediate reference, collocational pattern, indicating verbs and referential distance. But in case of Bengali, the indicating verb indicator has no role in filtering the antecedents and hence removed. Note that in some cases the verb has an important role (Chapter 6) to resolve conflict in case of multiple candidate antecedents and therefore, the verb transitivity information has been used in our implementation.

4.4 Evaluation

The configured GuiTAR system has been evaluated with the ICON data [14] for Bengali. The data consists of nine texts from different domains (Tourism, Story, News article, Sports). We have extended the data set by adding four more texts in the same format. Among these four pieces, three are short stories and one is a newspaper article. Table 4.4 shows the distribution of pronouns in the whole test data set for Bengali.

Table 4.4: Coverage of the Test Data

Data	ICON2011	Extended
# text	9	4
# words	22,531	4,923
# pronouns	1,325	322
# anaphoric	1,019	253

The dataset contains 1,647 pronouns out of them 706 are personal pronouns (including reflexive pronouns). As the MARS in GuiTAR resolves only personal pronouns, we have only used these personal pronouns for evaluation. Three different systems are configured as described below:

System-1 (Baseline): A baseline system is configured by considering the most recent noun phrase as the referent of a pronoun (the first noun phrase in the backward direction is the antecedent of a pronoun).

System-2 (GuiTAR with MARS): In this configuration, GuiTAR is used with the modifications (to build MAS-XML) in its pre-processing module and the modified MARS (in the algorithm) is used for pronominal anaphora resolution (PAR).

System-3 (GuiTAR with a new PAR module): Under this configuration, GuiTAR is used with the modifications (for Bengali) in its pre-processing module, but MARS is replaced by a previously developed system ([14]) for pronominal anaphora resolution in Bengali. This is basically a rule-based system. For every noun phrase (i.e. a possible antecedent) the method first maintains a list of possible pronouns which the antecedent could attach with (note that any noun phrase cannot be referred by any pronoun). On encountering a pronoun, the method searches for the antecedents for which the pronoun is in the respective pronoun-list. If there is more than one such antecedent, a set of rules is applied to resolve. The approach for applying the rules is similar to the one proposed by Baldwin [16]. The finer details of this algorithm is presented in the next chapter.

4.4.1 Evaluation Metrics

An important problem in co-reference/anaphora resolution is how to evaluate a system's performance. According to Luo [17], a good performance metric should have the following two properties: (i) Discriminative property: This refers to the ability to differentiate a good system from a bad one. While this criterion sounds trivial, not all performance metrics used in the past possessed this property. (ii) Interpretability: A good metric should be easy to interpret. That is, there should be an intuitive sense of how good a system is when a metric suggests that a certain percentage of co-reference results are correct. For example, when a metric report 95% or above accuracy for a system, we would expect that the vast majorities of the mentions are in right entities or co-reference chains. But none of the single metric satisfies these properties properly. And hence we have used standard measures for anaphora resolution, i.e. precision, recall and F-measure. These measures are calculated in five different perspectives, namely, MUC [18], B^3 [19], CEAFM [20], CEAFE [20] and BLANC [21] which are briefly explained below.

MUC-score: This metric is a linked based measures i.e. "the scheme operates by comparing the equivalence classes defined by the link's in the key and the response, rather than the links themselves" [18]. Recall and precision are calculated based on the links between the true chains and the system's output chains. Recall is computed as the number of common links between the true chains and the system's output chains with respect to the number of links in the true chains. Whereas, precision is computed as the number of common links with respect to the number of links in the system's output chains.

B-cubed (B^3): There are two shortcomings in MUC measure [19]. The first one is that, only gain points for links. No points gained for correctly recognizing that a particular mention is not anaphoric. And the second one is all errors are equal. In B^3, instead of looking at the links produced by a system, here we consider the presence or absence of entities relative to each of the other entities in the equivalence classes produced. To compute B^3 one needs to compute the recall and precision for each mention as follows.$Precision_i$ = *(number of correct elements in the output chain containing entity*$_i$*)/(number of elements in the output chain containing entity*$_i$*) and* $Recall_i$ = *(number of correct elements in the output chain containing entity*$_i$*)/(number of elements in the truth chain containing entity*$_i$*).* And to obtain the overall recall and precision, an average is taken over recall and precision values.

CEAF score: Luo [17] criticized the B^3 measure for using entities more than one time, because B^3 computes precision and recall of mentions by comparing entities containing that mention. Hence Luo proposed the CEAF measure based on an alignment score between the key and response entities and used the Kuhn-Munkres graph matching algorithm. Luo's definition resulted in mention-based (CEAFM) and entity-based (CEAFE) evaluation metrics.

BLANC score: "The BLANC [21] algorithm deals correctly with singleton entities and rewards correct entities according to the number of mentions. However, a basic assumption behind BLANC is that the sum of all co-referential and non co-referential links is constant for a given set of mentions. This implies that BLANC assumes identical mentions in key and response".

4.4.2 Results

The experimental results are reported in Table 4.5. Results show that GuiTAR with MARS gives better result than the situation where the most recent antecedent is picked (i.e. the baseline system). This improvement is statistically significant ($p < 0.03$ in a two-sided t-test). When MARS is replaced by System-3, further improvement is achieved, which is also statistically significant ($p < 0.01$).

Table 4.5: Experimental results

System		Baseline	GuiTAR	
Matric			System2 (MARS)	System3
MUC	P	0.453	0.516	0.538
	R	0.550	0.536	0.579
	F1	0.497	0.526	0.558
B^3	P	0.766	0.828	0.921
	R	0.771	0.824	0.911
	F1	0.769	0.826	0.916
CEAFM	P	0.785	0.800	0.885
	R	0.632	0.622	0.784
	F1	0.700	0.700	0.832
CEAFE	P	0.797	0.825	0.921
	R	0.552	0.571	0.731
	F1	0.652	0.675	0.815
BLANC	P	0.688	0.700	0.732
	R	0.735	0.736	0.741
	F1	0.711	0.718	0.736
Avg.	F1	0.666	0.689	0.771

4.5 Summary

This chapter presents a pioneering attempt to reconfigure GuiTAR (originally developed for English) for Bengali. The system, i.e., GuiTAR has been studied first at length and the language specific issues are identified to reconfigure it for Bengali. Such issues are broadly classified into two categories: pre-processing and language dependency of the resolution protocol. In pre-processing pipeline GuiTAR creates the MAS-XML which contains all the necessary annotations required for anaphora resolution. We resolved the issue by creating the MAS-XML externally from our tagged data [14]. In resolution algorithm, the gender agreement has been used in original GuiTAR and this agreement has no role in Bengali, instead the honorific agreement used for Bengali. This is resolved by removing the gender agreement and introducing the honorific agreement in the implementation. Finally, the system is evaluated by the publicly available data [14] which is further extended at our lab.

Chapter 5

Pronoun Emitting Approach for Anaphora Resolution in Bengali

5.1 Introduction

This chapter describes a rule based domain independent Pronominal Anaphora Resolution (PAR) system for Bengali. Philosophically the method is different from the other approaches. The main idea of this approach is that, a noun can't be referred by any pronoun of that language. Rather, the noun can associate with a limited number of pronouns and varies from one noun category to the other. For example, the noun **Taj Mahal** (a tomb) or **Kolkata** (a city) cannot associate with the pronoun আমি/ *Ami* (I), similarly the pronoun এটা/ *eTA* (it) can't be associated with the noun **Narendra Modi** (a person). In all the existing approaches, a pronoun goes backwards to find its antecedent, but in this approach, a noun (possible antecedent) in a discourse emits its permissible pronouns and finds its co-referent (pronoun) in the forward direction in the discourse. For more than one candidate antecedent, i.e. in case of conflict, a syntactic agreement based conflict resolution module has been designed. Initially, this system [22] was presented in ICON2011 NLP tool contest on Anaphora Resolution in

Indian Languages [14] and the result of the system won the second position in Bengali language. Here we have presented an improved and extended version of that system. In this approach a *language recourse* has been used. The subsequent sections describe the language recourse and the algorithm in details. The approach has been tested on publicly available (ICON2011 NLP tool contest data) data set [14] (the dataset has been further extended by us). Finally the result has been compared with the other existing system with a common data set and our result has proved the superiority of the present approach.

5.2 Motivation

The share task on anaphora resolution in Indian languages was initiated by ICON 2011 [14] under the track: "Anaphora Resolution in Indian languages, the tool contest". The brief descriptions of the systems participated are presented in Chapter 1 [1.4.3]. The contest had three languages, namely, Hindi, Bengali and Tamil. All the participants submitted their results in Bengali. We participated for Bengali and our system got the second position [22]. This share task enlightens us about the current research status of Anaphora Resolution in Bengali and other Indian languages and the limitations in this research field. This situation motivated us to develop a better and robust Anaphora Resolution system for Bengali and finally we have improved and extended our work presented in ICON 2011.

5.3 The Linguistic Resource

In Chapter2 [2.1], the corpus based analysis of Bengali pronoun gives some valuable information and features about pronoun. Though many of these features and exceptions [2.3] were discussed in linguistics literature but they were are not incorporated in anaphora resolution system. The statistical analysis shows the relative frequency of pronouns (Table 2.4) and most frequent pronouns are personal pronoun and this information helps to develop a practical anaphora resolution system. As said earlier that Bengali is a resource constraint language and no suitable morphological analyser is available especially for Bengali pronoun. So we have designed a **Language Resource** (LR) in order to bypass the requirement of NLP tools for pronominal anaphora resolution. The LR can be considered an important component in our system. Our LR is shown in the Table 5.1, it is not only the collection of significant grammatical information of pronouns but also, it has some other language specific information which is used to resolve anaphora. The detailed description of this resource is given below.

Chapter 2 [2.4] shows that, there is a large number of pronouns (and their inflectional forms) exist in Bengali. For pronominal anaphora resolution it is very complicated to handle such a large number of pronouns and their behavioural variations. To make it manageable the pronouns are categorized i.e. grouped according to their syntactic features together with the other grammatical features (i.e. classifier, case marker, etc.). The pronouns are classified (done manually by linguistic experts) based on the person (first person, second person, third person), number (singular, plural), animate, inanimate, reflexive pronoun, the locative pronoun (pronouns having the referent as location), quantifier pronoun (pronouns having the referent as quantity), etc. Personal pronouns are also classified based on honorific information with different degrees of honour [2.3.9]. Apart from the pronoun classification, LR also contains an additional information as follows.

Honorific Addressing terms : The honorific addressing terms in Bengali like বাবু/ *bAbu*, ডঃ/ *DaH* (Dr.), মহাশয়/ *mahAshay* (sir), etc. convey honorific information of a person in Bengali. These terms co-occur either before or after the name of a person.

Nominal relations : Collection of nominal relations like মা/ *mA* (mother), বাবা/ *bAbA* (father), ভাই/ *bhAi* (brother), বোন/ *bona* (sister), দাদা/ *dAdA* (elder brother), etc. in Bengali.

Possible common noun antecedents : Generally the data used in anaphora resolution is annotated with POS tags, named entity information, chunking information, etc. Our test data [5.6] annotated with several information, .e.g., all proper nouns (NNP) and some common nouns (NN) are tagged with PERSON, LOCATIONS, or ORGANIZATION, etc. But many common nouns (NN) may be the antecedent but are not tagged with NE classes. The collection of such common nouns (for PERSON and LOCATION categories) are stored in LR (Table 5.1).
Singular marker : The markers টি/ *Ti*, টা/ *TA*, খানি/ *khAni*, খানা/ *khAnA*, etc. are used to identify the singularity of the noun.
Plural marker : The markers রা/ *rA*, দের/ *dera*, গুলি/ *guli*, গুলো/ *gulo*, etc. are used to identify the plurality of the noun.

This resource has been built manually by the categorization of the pronouns found in Chapter 2 from the TDIL corpus [1] and FIRE data[2]. Linguistic experts were hired to develop this resource. The content of the LR is addresses all the pronouns from the corpus (about 800 pronouns), suffixes (90 suffixes), honorific terms (28 addressing terms), co-occurring pronouns (47 pair), connectors (12), etc. Note that though the total number of pronouns (Table 2.2) is more than 1,500 but we have considered the most frequent 800 pronouns from the corpus.

5.4 Proposed method

In all existing approaches, pronoun goes backwards to find its antecedent, but in our approach, a noun (possible antecedent) emits its permissible pronouns from linguistic resource to form an antecedent object and goes forwards to find its co-referent (pronoun). The Figure 5.1 shows the pictorial representation of our pronoun emitting resolution

[1] http://tdil.mit.gov.in/

[2] http://fire.irsi.res.in/fire/2016/home

Table 5.1: The Language Resource (LR)

CATEGORY	Permissible Values
Singular	আমি/*Ami*(I), তুমি/*tumi*(you), সে/*se* (he) ...
Plural	আমরা/*AmarA*(we), তোমরা/*tomarA*(you), তোরা/*torA* (you) ...
1st Person	আমি/*Ami*(I), আমার/*AmAra*(my), মোর/*mora*(my) ...
2nd Person	তুমি/*tumi*(you), তুই/*tui*(you), তোর/*tora*(your) ...
3rd Person	এর/*era*(his), সে/*se*(he), উনি/*uni*(he) ...
Animate	আমি/*Ami*(I), তুমি/*tumi*(you), সে/*se*(he) ...
Inanimate	ওটা/*oTA*(that), এটা/*eTA*(this), সেটা/*seTA*(that) ...
Locative Pronoun	যেখানে/*yekhAne*(where), এখানে/*ekhAne*(here) ...
Quantifier Pronoun	এত/*eta*(so much), তত/*tata*(that much) ...
Reflexive Pronoun	স্বয়ং/*sbayM*(self), নিজে/*nije*(self), নিজের/*nijera*(own) ...
Honorific Singular	উনি/*uni*(he), ইনি/*ini*(he), আপনি/*Apani*(you) ...
Honorific Plural	তাঁদের/*tA.Ndera*(them), উনারা/*unArA*(they), ...
Co-occurring Pronoun	যখন/*yakhana*(when), তখন/*takhana*(that moment) ...
Relational Term	মা/*mA*(mother), বৌ/*bau*(wife), বাবা/*bAbA*(father) ...
Connector	ও/*o*(and), আর/*Ara*(and), বা/*bA*(or) ...
Honorific Addressing Term	শ্রী/*shrI*(Mr.), ডঃ/*DaH*(Dr.), মিঃ/*miH*(Mr.), বাবু/*bAbu*(sir) ...
Common noun antecedent	লোক/*loka*(person), মেয়ে/*meye*(girl), বধূ/*badhU*(wife/bride) ...
Community	মজুর/*majura*(worker), কৃষক/*kRRiShaka* (farmer) ...
Singular marker	টি/*Ti*, টা/*TA*, খানি/*khAni*, ...
Plural marker	রা/*rA*, দের/*dera*, গুলি/*guli* ...
....	....

process and comparisons with the existing approaches.

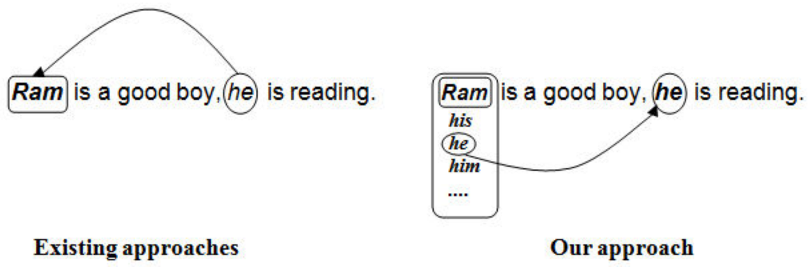


Figure 5.1: The pronoun emitting approach in comparison with the other existing methods

The pronominal anaphora resolution (PAR) system consists of a rule base (ten rules) along with the pronoun emitting part. In case of a conflict, i.e, when more than one possible antecedent/co-referent exists, a separate component the conflict resolution module is invoked. The formal algorithm, system architecture, rule base and the pronoun emitting approach are presented in subsequent sections. The notations used in the PAR algorithm are defined in Table 5.2. The system architecture is shown in Figure 5.2 and the system details are described by the algorithms Algorithm 3 - Algorithm 6.

Table 5.2: The notations used in the PAR algorithm

w_i	=	current input token
t	=	NE tag of token (noun) // Example: PERSON, LOCATION, ...
P	=	$\{p_1, p_2, ...\}$, pronoun set corresponds to a particular NE tag
a_i	=	antecedent object, // details given in 5.5
	=	possible antecedent with its permissible pronoun set
	=	[w_i: P]; // Example: [London: {where, there, ...}]
L	=	antecedent object list // Example: $\{a_1, a_2, ...\}$
d(a_i)	=	sentence level distance of a_i from w_i
LR	=	linguistic resource // Table 5.1
NNP	=	proper noun
NN	=	common noun
PRP	=	pronoun

```
Begin
    L = { }; /* initially empty */
    $w_i$ ← next input;
    if $w_i$ == NNP or NN then /* annotated in data */
        $a_i$ = createAntecedentObject (); /* Algorithm 4 */
        L = L ∪ {$a_i$}; /* add $a_i$ in L */
    else if $w_i$ == PRP then
        handlePronoun(); /* Algorithm 5 */
    else
        do nothing; /* skip */
    end if
End
```

Algorithm 3: PAR: Pronominal Anaphora Resolution algorithm

5.4.1 Rule base generation

As mentioned in Chapter 1 [1.4.4], Sengupta [23] and Majumder [24] have proposed some rules based on the linguistic analysis of pronouns. They explained all such rules based on particular examples, but not in a generic way. In our system a rule extraction technique has been followed. The approach is based on the high coverage and high satisfiability of a rule in the data set. The high coverage implies that the number of instances present

```
Begin
    else if $w_i$ == NNP then
        $t$ = getDescription(); /* given in data */
        $P$ = emits permissible pronoun set from LR based on $t$; /* Rules (1-2) are applied */
        $a_i$ = [$w_i$: P]; /* create antecedent object */
    else
        if $w_i$ == NN then
            $t$ = getDescription(); /* if not given in data then get it from LR */
            if $t$ found then /* description available in LR */
                $P$ = get compatible pronoun set from LR based on $t$;
                $a_i$ = [$w_i$: P]; /* create antecedent object (Table 5.2) */
            end if
        end if
    end if
End
```

Algorithm 4: createAntecedentObject(): Algorithm for Create Antecedent Object

```
Begin
    if $p_i$ == non anaphoric then /* Rules (3-5) decide whether $p_i$ is non-anaphoric */
        do nothing; /* skip */
    else
        if solved by direct rule then /* Rule (6-10) */
            update $L$; /* create antecedent object for pronoun $p_i$ (in Algorithm 2) and add in $L$ */
            print result; /* resolved */
        else
            $A = \{\forall\ a_i = [w_i\text{: P}]\text{: } p_i \in P\ \&\ d(a_i) \leq 2\ \}$ /* find possible antecedents within 2 sentences */
            if |A| == 1 then
                update $L$; /* add $a_i$ in $L$ */
                print result; /* resolved */
            else if |A| > 1 then
                conflictResolution(); /* Algorithm 6 */
            end if
        end if
    end if
End
```

Algorithm 5: handlePronoun(): Algorithm for Pronoun Resolution

```
Begin
    $A$ /* possible antecedents */
    if $p_i$ is person compatible with unique $a_i \in A$
        update $L$; /* add $a_i$ in $L$ */
        print result; /* resolved */
    else if $p_i$ is number compatible with unique $a_i \in A$ then
        update $L$; /* add $a_i$ in $L$ */
        print result; /* resolved */
    else if $p_i$ is honorific compatible with unique $a_i \in A$ then
        update $L$; /* add $a_i$ in $L$ */
        print result; /* resolved */
    else choose the most recent $a_i \in$ then /* with minimum $d(a_i)$ */
        update $L$; /* add $a_i$ in $L$ */
        print result; /* resolved */
    end if
End
```

Algorithm 6: conflictResolution(): Algorithm for Conflict Resolution

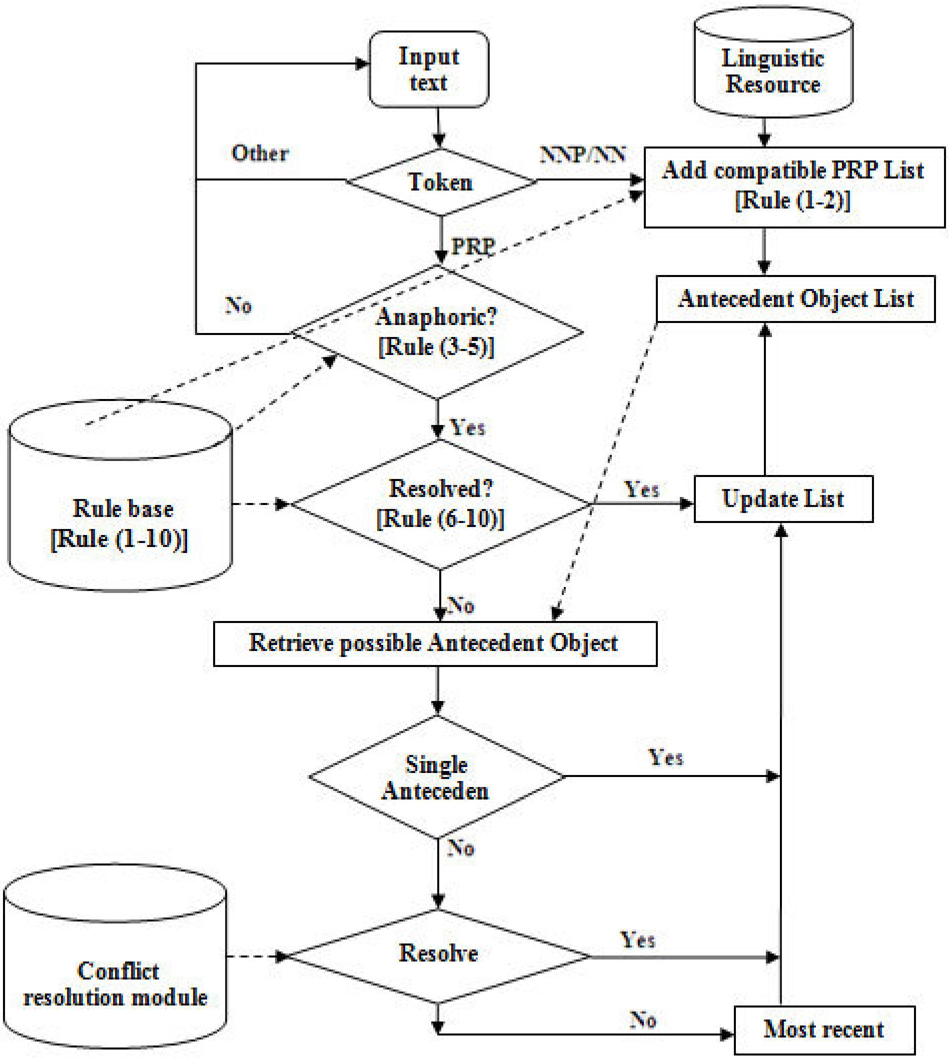


Figure 5.2: Architecture of the PAR system

in the corpus is statistically significant and high satisfiability implies that high precision and high recall. Following this criterion we generalize some previously found rules and then tested and accepted in our study. Moreover, some new rules have been found those are not explored earlier. Most of the rules have been extracted using syntactical information like part-of-speech, case, person, number, particle, honorific information etc.

5.4.1.1 Rule base

As said earlier our rule base contains ten rules [5.4]. The rules can be categorized into three types based on their functionality. The first category (Rule-1 - Rule-2) is used to identify the syntactic information (identifying number and honorific information) of possible antecedents. The second category (Rule-3 - Rule-5) used to identify the non anaphoric pronoun and finally the third category (Rule-6 - Rule-10) is used for direct resolution of an anaphora.

5.4.1.2 Rules for identifying syntactic information

Rule-1: Identification of number information
A noun is singular if it is inflected with singular classifier (Ø (null), টি/Ti, টা/TA, -খানি/ *khAni*, খানা/ *khAnA*, ...) and a noun is plural if it is inflected with plural classifier (রা/ *rA*, দের/ *dera*, গুলি/ *guli*, গুলো/ *gulo*, ...) (Classifiers are given in Table 5.1).
Note that, an organization, a community or collective noun are considered as plural.
Rule-2: Identification of honorific information
The person addressing with the honorific addressing terms (i.e. ডঃ/ *DaH* (Dr.), শ্রী/ *shrI*, প্রঃ/ *praH* (prof.), ...) immediately before the name or the terms (সাহেব/ *sAheba*, মহাশয়/ *mahAshay* (sir), দেবী/ *debI*, বাবু/ *bAbu*, মশাই/ *mashAi*(sir), ...) immediately after the name (in Table 5.1) or having the inflection ন/ *n* of the main verb is consider as honorable person. Note that, we also found the honorific information using the maximum entropy classifier [254] and imported the result in the resolution process and we didn't observe any significant improvement.

5.4.1.3 Rules for identifying non-anaphoric pronouns

Rule-3: Reduplicated pronoun
Reduplicated pronouns i.e. same pronoun occurring twice consecutively are non-anaphoric. Such as, [কিছু কিছু/ *kiChu kiChu*], [নিজের নিজের/ *nijera nijera*], [কেউ কেউ/ *keu keu*], [কারও কারও/

kArao kArao], [কখনো কখনো/ *kakhano kakhano*], etc.

Example: আবার জাতীয় ক্রিকেট মহলে [কেউ কেউ] মনে করিয়ে দিতে চান ... / *AbAra jAtIya krikeTa mahale* [*keu keu*] *mane kariyae dite chAna* ...(In the national cricket association some people want to remind ...).

Exception: The rule is not applicable in case of repetition of interrogative pronoun i.e. কী কী/ *kI kI* (plural form of what), কে কে/ *ke ke* (plural form of who) etc.)

Rule-4: Inherently non anaphoric

In the corpus, some pronouns (তা/ *tA*, তেমন/ *temana*, যা/ *yA*, কেউ/ *keu*, ...) are identified that they rarely behave as anaphoric and we define such pronouns as inherently non anaphoric.

Example: চৌত্রিশ নম্বর আশ্রয় যে না থাকতে পারে , সুলেখার পক্ষে [তা/PRP] বোধ হয় অকল্পনীয় ছিল । /*chautrisha nambara Ashra ye nA thAkate pAre , sulekhAra pakShe* [*tA*/PRP] *bodha hay akalpanIy Chila* (It was unbelievable to Sulekha that there was no shelter at address, thirty four). In this example, the pronoun তা/*ta* (that) is not anaphoric.

Example: এখানে [যে/PRP] আমি চুপি চুপি সরে এসেছি , [তা] কাকপক্ষী পর্য্যন্ত না জানলেই সুবিধে । /*ekhAne* [*ye*/PRP] *Ami chupi chupi sare eseChi ,* [*tA*/PRP] *kAkapakShI paryyanta nA jAnalei subidhe* (It will be better, if no one knows that I have come here silently).

In the second example, the first pronoun যে/*je* is non-anaphoric but the second pronoun তা/ *ta* refers the entire part of sentence "এখানে যে আমি চুপি চুপি সরে এসেছি"/ *ekhAne ye Ami chupi chupi sare eseChi.* For the time being we are not considering তা/*tA* as anaphoric.

Rule-5: Non anaphoric reflexive pronoun

A sentence starting with reflexive pronoun is not anaphoric.

Example: [নিজের/PRP] শরীর দুর্বল হয়ে যাচ্ছে বুঝলেও অনিমেষ মুখে প্রকাশ করেনি । /[*nijera*/PRP] *sharIra durbala ha e yAchChe bujhaleo animeSha mukhe prakAsha kareni* (Animesh has not expressed anything even after knowing that he is becoming weak). In this sentence, the pronoun নিজের/ *nijera* (own) is non-anaphoric (here the referent of নিজের/ *nijera* (own) is in forward direction, it is example of cataphora).

5.4.1.4 Rules for identifying co-referent pairs

Rule-6: Rule for second person singular pronouns

The second person singular pronouns are তুমি/ *tumi* (you) and তুই/ *tui* (you) (non honorific form of "you").

Case-1: If a NNP or NN (person) is inflected with (কে/ *ke*, কেউ/ *keu*, টিকে/ *Tike*, র/ *ra*) and followed by a second person singular (তুমি/ *tumi*) pronoun then the pronoun does not co-refer with that NNP of NN; otherwise তুমি/ *tumi* co-refers with that NNP of NN.

Example: a) [সীমা/NNP], [সুলেখাকে/NNP] [তুমি/PRP] বিদায় দাও। / [*sImA*/NNP], [*sulekhAke*/NNP] [*tumi*/PRP] *bidAy dAo* ([*Sima*/NNP], [*you*/PRP] *please bid good bye* to [*Sulekha*/NNP]). In this example, [তুমি/PRP] does not co-refer with [সুলেখাকে/NNP]. Since, NNP with inflection with কে/ke followed by তুমি/ *tumi* pronoun.

b)[সন্ন্যাসীদাদা/NN_PERSON], [তুমি/PRP] ত সত্যিই সন্ন্যাসী নও তোমার শরীরে দয়া মায়া আছে। /[*sannyAsI-dAdA*/NN_PERSON], [*tumi*/PRP] *ta satyii sannyAsI nao tomAra sharIre dayA mAyA AChe* ([Sanyashidada/NN_PERSON], [you/PRP] are not really monk, you have worldly emotion). In this example [তুমি/PRP] is co-referring with [সন্ন্যাসীদাদা/NN_PERSON] (Since NNP followed by তুমি/*tumi* pronoun).

Case-2: If a NNP or NN (PERSON) is followed by comma (,) followed by second person singular pronoun (তুমি/tumi) then the pronoun co-refers with that NNP of NN (PERSON).

Example: আচ্ছা বেশ, পাঠাচ্ছি [মা/NN_PERSON , তুমি/PRP] বেশি ভেবো না। / *AchChA besha, pAThAchChi* [*mA*/NN_PERSON] , [*tumi*/PRP] *beshi bhebo nA* ([Mom/NN_PERSON], it is fine, I am sending, [you/PRP] do not worry).

Exception: তোমার মা ছিলেন বিক্রমপুরের [মেয়ে/NN_PERSON], [তুমি/PRP] নদীয়া জেলার ভদ্রসন্তানকে বাঙাল .../*tomAra mA Chilena bikramapurera* [*meye*/NN_PERSON], [*tumi*/PRP] *nadIyA jelAra bhadrasantAnake bA∼NAla* ... (Your mother came from Bikrampur but [you/PRP] are addressing a gentleman from Nadia district as *Bangal*).

Rule-7: Rule for consecutive pronoun

Two consecutive personal pronouns are co-referring if both are first person or both are second person. Otherwise (permutation of first and second person) they are not co-referent. (Note that, this rule is not applicable in case of third person pronoun).

Example: [আমি/PRP আমার/PRP] সব কাজ নিজে করি । /[*Ami*/PRP *AmAra*/PRP] *saba kAja nije kari* ([I/PRP] [myself/PRP] do all my work). The consecutive pronouns আমি/PRP and আমার/PRP are both in first person and hence co-refer.

Example: [তুমি/PRP তোমার/PRP] কাজ কর । /[*tumi*/PRP *tomAra*/PRP] *kAja kara* ([You/PRP do [your/PRP] own work). The pronouns তুমি/PRP and তোমার/PRP both is second person and hence co-refer.

Example: [আমি/PRP তোমাকে/PRP] ভালবাসি । /[*Ami*/PRP *tomAke*/PRP] *bhAlabAsi* ([I/PRP] love [you/PRP]). The pronouns আমি/PRP is in first person but তোমাকে/PRP is in the second person and hence do not co-refer.

Rule-8: Rule for reflexive pronoun

Case-1: If NNP/NN/PRP (human) (not inflectional with কে/ke) is followed by reflexive pronoun then this NNP/NN/PRP is the antecedent of the reflexive pronoun. Otherwise, the NNP/NN/PRP is not co-referring with the reflexive pronoun.

Example: [শিখাদেবী/NNP] [নিজের/PRP] যোগ্যতায় নয়, কোটার টিকিট পেয়েছিলেন ! /[*shikhAdebI*/NNP [*nijera*/PRP] *yogyatAya naya, koTAra TikiTa peyaeChilena* ([Sikhadebi/NNP] got the ticket under reserved category, not by her [own/PRP] talent). In this example, sikhadebi/NNP is the antecedent of nijer/PRP.

Example: যথাবিহিত চিকিৎসার পর শ্রীরামকৃষ্ণ [তাকে/PRP] [নিজের/PRP] শয্যায় শয়ন করার অনুমতিও দিয়েছিলেন। /*yathAbihita chikitsAra para shrIrAmakRRiShNa* [*tAke*/PRP] [*nijera*/PRP] *shayyAy sha na karAra anumatio diyeChilena* (After giving proper treatment Sriramkrihna asked [him/PRP] to sleep in [his/PRP] bed). In this example, pronoun তাকে/*tAke* (inflection with কে/*ke*) is not the antecedent of নিজের/PRP.

Case-2: If the antecedent is not found in Case-1, then choose the next most recent NNP/NN/PRP (PERSON) in backward.

Example: আমি নিশ্চিত, [হ্যারি/NNP] [তোমাকে/PRP] [নিজের/PRP] জীবনের চেয়েও বেশী ভালবাসে। /*Ami nishchita,* [*hyAri*/NN] [*tomAke*/PRP] [*nijera*/PRP] *jIbanera cheyeo beshI bhAlabAse* (I am sure that Harry loves you more than his own life). In this example, তোমাকে/ *tomAke* (inflectional with কে/ *ke*) is not the antecedent of নিজের/PRP. The antecedent is the most recent noun i.e. হ্যারি/hyAri/NNP.

Rule-9: Rule for relational term/connector

If there is a sequence like "PRP Relational_Term PRP" or "PRP Connector PRP" (LR gives the relational terms, and connectors).

Case-1: If both the PRPs are in the first person or in the second person then they must co-referring.

Example: আমি যখন শুতে চাই আর [আমার/PRP ভাই/ Relational_Term আমায়/PRP] বিরক্ত করে … /*Ami yakhana shute chAi Ara* [*AmAra*/PRP] [*bhAi*/Relational_Term] [*AmAy*/PRP] *birakta kare* … ([my/PRP] [brother/Relational_Term] disturbs [me/PRP] when I want to sleep....).

Case-2: If both the PRPs are not in the first person or both are not in the second person then they must not co-refer.

Example: স্নান করে শিবপূজা করছিলাম, [তোমার/PRP ছেলে/ Relational_Term আমার/PRP] শিবলিঙ্গটি নিয়ে পালিয়েছে৷ / *snAna kare shibapUjA karaChilAma* , [*tomAra*/PRP] [*Chele*/Relational_Term] [*AmAra*/PRP] *shibali∼NgaTi niye pAliyeChe* … (After taking a bath, when I was praying, [your/PRP] [son/Relational_Term] took away [my/PRP] Shivalinga).

Rule-10: Rule for co-occurring pronouns

In Bengali, when some pronoun pairs ([যাদের/yAdera, তাদের/tAdera], [যাঁর/yA.Nra, তাঁর/tA.Nra], [যেখানে/yekhAne, সেখানে/sekhAne], [যখন/yakhana, তখন/takhana] etc.) appear in the same sentence, they are co-referring. We define such pronoun pairs as co-occurring pronouns. In our study we have found 47 such co-referring pairs (Table 5.1).

Example: লক্ষ্মী [যেখানে/PRP] অধিষ্ঠিত, বিষ্ণুপ্রিয়া কেমন করে [সেখানে/PRP] স্থান পেতে পারে? /*lakShmI* [*yekhAne*/PRP] *adhiShThita, biShNupriyA kemana kare* [*sekhAne*/PRP] *sthAna pete* *pAre* ([Where/PRP] the goddess Laxmi is already seated, how can the goddess Visnupriya take place [there/PRP]?). Here the pronouns [যেখানে/PRP] and [সেখানে/PRP] are co-referring. Similarly, in the example: ... এবং [যেগুলির/PRP] তারকা-চিহ্নিত থাকবে না, [সেগুলির/PRP] হসন্ত উচ্চারিত হবে। /... *ebaM* [*yegulira*/RPP] *tArakA-chihnita thAkabe nA* , [*segulira*/PRP] *hasanta uchchArita habe* (... and the words having no star sign, will be pronounced as of Hasanta symbol), the pronouns [যেগুলির/PRP] and [সেগুলির/PRP] are co-referring.

5.5 Illustration of the Pronoun Emitting Approach (PEA)

This section illustrates the pronoun emitting approach with an example in details. Consider the sentence, "ব্যারেট/NNP বলে উঠলেন যে তিনিই/PRP বড় হাতীটাকে গুলি করবেন। "/ "*byAreTa/NNP bale uThalena ye tinii/PRP baDa hAtITAke guli karabena*", taken from *story2.txt* of ICON 2011 data. Note that, the actual annotated data is in the column format shown in Table 5.3.

The first part of this approach is the creation of antecedent object (Algorithm 2) i.e. by using the function *createAntecedentObject()* [5.3]. Note that, in annotated data (Table 5.6) POS of "ব্যারেট" is NNP with NE tag PERSON. The person "ব্যারেট/NNP" is identified as honourable person (by Rule-2), hence it emits the permissible pronoun list (Honorific singular i.e. list তাঁর, তাঁকে, তিনি, তিনিই, ...) from linguistic resource [5.1]. The schematic diagram for creation of antecedent object using Algorithm 2 is shown in Figure 5.3.

For computational point of view the antecedent object has other lexico-syntactic information like, unique id (Antecedent Object Id), honorific information, sentence id, token id, co-refer object id (id of the object that co-refer with the current entity), etc. The exact antecedent object for "ব্যারেট" is shown in Table 5.4. Later on, when a pronoun is found, its resolution considers only those antecedent objects containing that pronoun. In this example, when the system goes to resolve the pronoun "তিনিই" (line 4 in Table

Table 5.3: Sample data from story2.txt of ICON 2012 Data)

......					
0	ব্যারেট	NNP	B-NP	B-PERSON	-
1	বলে	VN	B-VGF	o	-
2	উঠলেন	VAUX	I-VGF	o	-
3	যে	CC	B-CCP	o	-
4	তিনিই	PRP	B-NP	o	-
5	বড়	JJ	B-NP	o	-
6	হাতীটাকে	NN	I-NP	B-PERSON	-
7	গুলি	NN	B-NP	o	-
8	করবেন	VM	B-VGF	o	-
9	।	SYM	I-VGF	o	-
......					

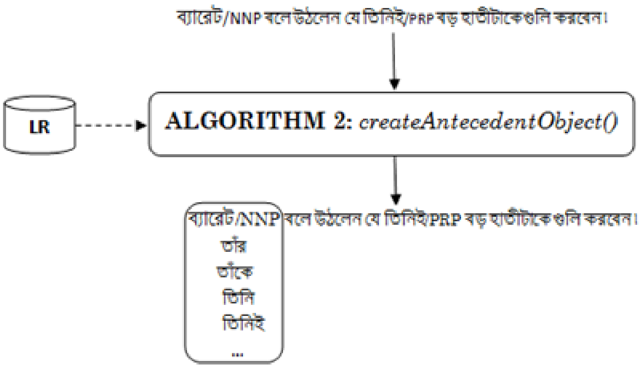

ব্যারেট/NNP বলে উঠলেন যে তিনিই/PRP বড় হাতীটাকেগুলি করবেন।
LR
ALGORITHM 2: createAntecedentObject()
ব্যারেট/NNP
বলে উঠলেন যে তিনিই/PRP বড় হাতীটাকে গুলি করবেন।
তাঁর
তাঁকে
তিনি
তিনিই
...

Figure 5.3: Creation of antecedent object for the person ব্যারেট/NNP

5.3), it is obvious that "ব্যারেট" is the possible antecedent, since the pronoun "তিনিই" is in the permissible pronoun set of "ব্যারেট". Actually this processing is not done in backward direction; instead the antecedent object finds its co-referent in forward direction. Pictorially it looks like as in Figure 5.4. In case of conflict (more than one antecedent) the conflict resolution module is used to resolve it.

In a similar manner, a pronoun also can be an antecedent object. Once a pronoun is resolved, then the resolved information (with the co-reference information) is also included in the antecedent object. In the above example, the pronoun "তিনিই" refers to the antecedent object "ব্যারেট" and hence, an antecedent object is created for "তিনিই" by Algorithm 6 and its description is given in Table 5.5. After resolution, the system also updates the referent information, i.e. includes its reference information. In this example, since “ব্যারেট” being the referent of “তিনিই”, hence the “Co-refer Object Id” field in Table 5.4 will be updated i.e. will be set to 4 (since the object id of “তিনিই” is 4). In this example, the system also generates one more antecedent object for “হাতীটাকে” with the pronoun list “Animate Singular” from LR in Table 5.1.

Table 5.4: Antecedent object of person ব্যারেট

Attribute	Value
Antecedent Object Id	3
Antecedent Name	ব্যারেট
POS	NNP
Description	PERSON
Permissible Pronouns	তাঁর, তাঁকে, তিনি, তিনিই, ...
Honorific Information	Honourable
Text Name	story2.txt
Sentence Id	12
Token Id	1
Co-refer Object Id	
Not Co-refer Object Id	
....	

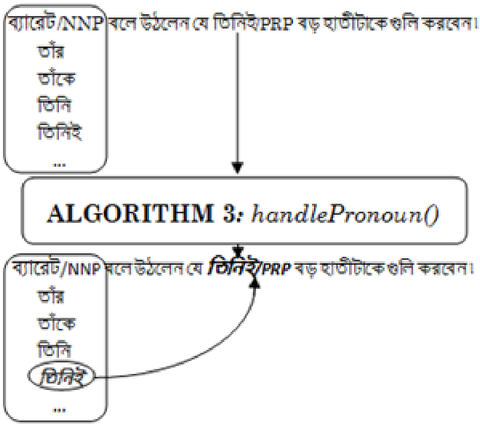

ব্যারেট/NNP বলে উঠলেন যে তিনিই/PRP বড় হাতীটাকে গুলি করবেন।
তাঁর
তাঁকে
তিনি
তিনিই
...
ALGORITHM 3: handlePronoun()
ব্যারেট/NNP বলে উঠলেন যে তিনিই/PRP বড় হাতীটাকে গুলি করবেন।
তাঁর
তাঁকে
তিনি
তিনিই
...

Figure 5.4: The antecedent of pronoun তিনিই

Table 5.5: Antecedent object of resolved pronoun তিনিই

Attribute	Value
Antecedent Object Id	4
Antecedent Name	তিনিই
POS	PRP
Description	
Permissible Pronouns	
Honorific Information	Honourable
Text Name	story2.txt
Sentence Id	12
Token Id	15
Co-refer Object Id	3
Not Co-refer Object Id	
....	

5.6 Experimental Setup

To evaluate the system, the data set provided by ICON 2011 [14] has been used. They provided annotated data with the information like POS tag, chunks and NE tag. The annotated data is represented by a column format. Table 5.6 shows a sample of the annotated data and the detailed description of the data is given in Table 5.7.

The ICON 2011 data contains nine texts from different domains (Tourism, Story, News article, Sports). We have extended this data set by adding four more texts in the same format. Among these four pieces, three are short stories and one is taken from newspaper articles. Table 5.8 shows the size of the data set and distribution of pronouns in the data set.

5.7 Evaluation

The system has been evaluated by the data set as described above. The evaluation is done in two phases, in the first phase the performance of the rules has been evaluated and in the second phase the performance of PEA system has been evaluated.

The validation and the coverage of the rules in the data set are shown in Table 5.9.

Table 5.6: ICON 2011 data format

story2.txt	0	0	সবশেষে	NN	B-NP	o	-
story2.txt	0	1	তার	PRP	B-NP	o	(22)
story2.txt	0	2	মনে	NN	B-NP	o	-
story2.txt	0	3	হলো	VM	B-VGF	o	-
story2.txt	0	4	এদিকে	NN	B-NP	o	-
story2.txt	0	5	আর	QF	B-NP	o	-
story2.txt	0	6	কোথাও	PRP	I-NP	o	-
story2.txt	0	7	বাঘ	NN	B-NP	B-LIVTHINGS	-
story2.txt	0	8	নেই	VM	B-VGF	o	-
story2.txt	0	9	।	SYM	I-VGF	o	-
....							

Table 5.7: Description of test data

Column	Type	Description
1	Document Id	Contains the file name
2	Part number	File are divided into part numbered
3	Word number	Word index in the sentence
4	Word	Word itself
4	POS	POS of the word
5	Chunking	Chunking information using IOB format
6	NE tags	Name Entity Information is given
7	Description	Description
8	Co-reference	Co-reference information

Table 5.8: Coverage of dataset

Data	ICON2011	Extended
#text	9	4
#words	22,531	4,923
#sentence	2,032	375
#pronouns	1,325	322
#anaphoric	1,019	253

Note that, since the Bengali language is the free order language, hence the exception can be found in some cases. But the number of such exceptions is negligible as evident from the statistics for satisfaction of the rules.

Table 5.9: Coverage and validation of the rules in test data

Rule	#Instance	#Correct	#Wrong	Precision	Recall	F-score
Rule-1	461	458	3	0.9934	0.9934	0.9934
Rule-2	157	156	1	0.9936	0.9936	0.9936
Rule (3-5)	277	274	3	0.9891	0.9891	0.9891
Rule-6	21	20	1	0.9523	0.9523	0.9523
Rule-7	36	36	0	1.0	1.0	1.0
Rule-8	23	23	0	1.0	1.0	1.0
Rule-9	5	5	0	1.0	1.0	1.0
Rule-10	23	22	1	0.9565	0.9565	0.9565

5.7.1 Comparison of PEA and GuiTAR

Performance of the pronoun emitting approach (PEA) has been evaluated using all the five metrics namely, MUC, B-cubed, CEAFM, CEAFE and BLANC as introduced in the previous chapter and the performance has been compared with the GuiTAR based pronominal anaphora resolution system on the same data set. We have done a comparison study of our system (PEA) with the previously configured GuiTAR system. Table 5.10 shows the result, i.e. the performance of the PEA system is relatively better than the configured GuiTAR system. We have investigated the reasons of improvement in PEA system over the GuiTAR-based system.

Table 5.10 shows the comparison of the GuiTAR-based system and our present PEA system. GuiTAR resolves pronouns using the MARS approach [8] that makes use of several agreements (based on person, number and gender). The GuiTAR-based system for Bengali has been designed by modifying the language specific features (like, removing the gender agreement and introducing honorific agreement).

The customized BART system [25] gives MUC score with recall 0.560; precision 0.465 and F1 score 0.50.8 on ICON 2011 test data set. Whereas, the average F1 score produced by the random tree algorithm in weka is 0.669 [26]. We compared the performance of the GuiTAR-based system with simple baseline system (last noun approach) and got the averages F1 score 0.666, and whereas GuiTAR gave 0.689. The results show that

Table 5.10: Comparison of experimental Result

System		GuiTAR	PEA
Matric			
MUC	P	0.516	0.583
	R	0.536	0.593
	F1	0.526	0.588
B^3	P	0.828	0.933
	R	0.824	0.921
	F1	0.826	0.927
CEAFM	P	0.800	0.898
	R	0.622	0.817
	F1	0.700	0.856
CEAFE	P	0.825	0.933
	R	0.571	0.781
	F1	0.675	0.850
BLANC	P	0.700	0.772
	R	0.736	0.781
	F1	0.718	0.776
Avg.	F1	0.689	0.799

GuiTAR or other customized approaches are not suitable for Bengali. Because the language dependent issues are not handled properly but the PEA system is relatively better than the other customized system. This improvement of PEA over GuiTAR is statistically significant ($p < 0.05$ in a two-tailed paired t-test). The details comparison and analysis is given next.

In principle, our system differs from GuiTAR system. Our system is based on a set of rules along with pronoun emitting technique. The GuiTAR system is implemented based on Mitkov anaphora resolution system [8], where, the decision is checked against the agreement and for a number of antecedent indicators. Candidates are assigned scores by each indicator and the candidate with the highest score is returned as the antecedent. The major drawbacks of GuiTAR in comparison to the present PEA system are explained below.

In PEA system, an organization, a community, etc. is considered as plural where the GuiTAR system considers them as singular and produces wrong results.

Example: "এদিকে [মাসাই কুলি/ORGANIZATION] [যারা/PRP] তাদের দৃষ্টি শুধুমাত্র ঐ তিনটে হাতীর ওপরেই ছিল।"/ *edike* [*mAsAi kuli*/ORGANIZATION] [*yArA*/PRP] *tAdera dRRiShTi shudhumAtra ai tinaTe hAtIra oparei Chila* (On the other hand, the Masai coolie, who concentrate on only those three elephants). In this case, PAR system resolves correctly for pronoun [যারা/PRP] with antecedent [মাসাই কুলি/ORGANIZATION] but GuiTAR system fails to resolve due to the mismatch of number agreement.

In PEA system, the Rules (3-5) identify the non anaphoric pronouns, but there is no way to identify the non anaphoric pronoun in the configured GuiTAR system. Hence GuiTAR system tries to resolve many non anaphoric pronouns and produces wrong results.

Example: "[কেউ/PRP] [কেউ/RDP] বলেন, চিমার মধ্যে ইদানিং অহংকার এসে আশ্রয় নিয়েছে।"/ [*keu*/PRP] [*keu*/PRP] *balena, chimAra madhye idAniM ahaMkAra ese Ashray niyeChe.* (Some people say that now a days Chima is suffering from ego). In this case, PEA system identifies correctly that the pronoun [কেউ/PRP] as non anaphoric (by Rule-3) but GuiTAR system resolves it wrongly.

The Rules (7-9) are highly confident (shown Table 5.9) and produce 100% accurate result and Rule 10 produces almost 100% accurate result and obviously, the use of these rules improves the performance of PEA system. In all such cases, the GuiTAR system finds antecedent based on assigned scores by many indicators and selects with the highest score and gets wrong result in many cases.

Example: "ভবিষ্যতে [যারা/PRP] সরকারের দায়িত্বপূর্ণ পদে যেতে চান [তারা/PRP] এসে ওর কাছে দিনরাত ধর্ণা দেনা"/ *bhabiShyate* [*yArA*/PRP] *sarakArera dAyitbapUrNa pade yete chAna* [*tArA*/PRP] *ese ora kAChe dinarAta dharNA dena* (Those who are interested to get important government positions in future, are appeasing her regularly). Here the PEA system identifies the co-referent correctly ([যারা/PRP] and [তারা/PRP]) by Rule-10 but GuiTAR system resolves it wrongly.

In case of conflict, the PEA system uses some agreement based rule to resolve conflict and if the conflict is still not resolved then heuristic approach (i.e. the most recent candidate chosen as the antecedent) is applied. But since GuiTAR system does not use such heuristic and hence gives correct result where the PEA system resolves wrongly in some cases.

Example: “[ক্ষিতিমোহন সেনের/PERSON] বাউলতৎত্ব আলোচনা ও লেখালেখিতে [স্টেন কোনোর/PERSON] খুবই উৎসাহ ছিল, [তাঁর/PRP] কাছে বাউলদের কথা শুনে ...”/ [*kShitimohana senera*/PERSON] *bAulatattba AlochanA o lekhAlekhite* [*sTena konora*/PERSON] *khubai utsAha Chila, tA.Nra kAChe bAuladera kathA shune …* (On hearing the Bawl story from Kshitimohon Sen, Stane Kone was highly interested in the discussion related to the Bawl flock …). In this example, the PEA system found the antecedent of [তাঁর/PRP] is [স্টেন কোনোর/PERSON] which is wrong, but GuiTAR system gives correct output as [ক্ষিতিমোহন সেনের/PERSON].

Now, from the above illustrations, it shows that on an average PEA system performs better than the configured GuiTAR system. This is also reflected in the experimental results given in Table 5.10.

5.7.2 Error analysis

In order to understand the weaknesses of our system an error analysis has been performed on the rule base and pronoun emitting phase. The goal of this analysis is to identify the major source of errors which influences the overall performance. The errors generated by the PEA system are broadly classified in the following categories.

Error in pre-processing phase: Errors are inevitably introduced at each pre-processing step, and these errors are reflected in the overall success of the system [27]. The pre-processing information is propagated in the subsequent steps and the effect of the error is multiplicative in nature. In our system, among the 10 rules, Rules (1-2) has been used for pre-processing. The Rule-2 is used to find the honorific information of a person locally i.e. within the sentence. But in Bengali the honorific information of a person may change and depends on local as well as global context. The rule does not consider such (global) scenario. Though the system identifies the honorific information locally, it uses this information globally.

Error in identifying non anaphoric pronouns: The Rule-3 to Rule-5 handle the non anaphoric pronouns. But these rules can handle only a small subset of actual non-anaphoric pronouns. Many pronouns appear anaphoric as well as non-anaphoric and it depends on the context and these cases are not tackled by the system.

Missing Antecedents: If pro-drop (a drop of the antecedent) or antecedents are located in long distant backward in the text, the system sometimes gives erroneous results.

Error in heuristic approach: When the system is not able to resolve conflict successfully, the system (conflict resolution module) chooses the most recent one (nearest antecedent) but in many cases it does not give correct results.

5.8 Summary

This chapter presents a novel approach for pronominal anaphora resolution (PAR) in Bengali. In most of the existing PAR approaches, the dominant trend is to look for the right antecedent after a pronoun is encountered. Unlike this trend the current PEA system follows a unique approach. On encountering a noun phrase (which could be a possible antecedent) this approach emits a list of permissible pronouns (note that a noun cannot associate with all possible pronouns). Next, when a pronoun is actually encountered, list(s) containing this pronoun is searched. The noun phrase that emitted this list emerges as the right antecedent. If the pronoun does exist in more than one list, a conflict resolution protocol is executed to find out the right antecedent. A set of rules is used along with the emitting approach. The rules are also evaluated and the result shows a very high degree of confidence. The system is evaluated with the publicly available data set and comparison is done with the previously configured systems. An error analysis is done and some issues are identified for future improvement.

Chapter 6

Some Advanced Issues for Anaphora Resolution in Bengali

6.1 Introduction

Though there is a considerable amount of research has been carried out in anaphora resolution over the last three decades, but still there is a number of outstanding issues present in this domain. Mitkov has already discussed several such issues [28]. In this chapter, we will address some of these issues with respect to Bengali language. We mainly focused on the issues related to pro-drop, one-anaphora and role of the verb in anaphora resolution and these issues are not explored earlier in Bengali. These issues are not only important in anaphora resolution, but also important for other applications like machine translation, information retrieval, question answering etc. Though, Bengali is not highly pro-drop language such as Japanese, Chinese, Italian, etc. but we have seen that in error analysis section in Chapter 5 some errors are due to the pro-drop in the language. Hence, to develop a high yield anaphora resolution system one has to consider this issue. Here in section 6.2 we have done the analysis of pro-drop and pro-drop resolution in Bengali. Most of the existing works in anaphora refer to the pronominal anaphora resolution because of the widespread use of pronouns. But there are other types of anaphora [8] also. One-anaphora is one of the important anaphora and it is present in many languages. In this chapter, we also concentrated on one-expression analysis in the corpus and it is discussed in the section 6.3. Finally, section 6.4 has addressed the issue related to the role of verbs using which anaphora can be resolved unambiguously in some cases.

6.2 Pro-drop in Bengali

The term **pro-drop** comes from (which is short for **pronoun dropping**) the lecture on Government and Binding by Noam Chomsky [29]. In many 'pro-dropped' languages often the pronouns are unrealized in the text. Such unrealized pronouns are regarded as zero anaphora, which are indicated using Φ in the literature. In our work, we considered the generic nature of drops in Bengali. Bengali is a pro-drop language and the dominant drops referring to the subject or object but sometimes verb drops also realised. For example:

- কখন এলেন?/ *kakhana elena?* (when did *you* come?).

In this example, the subject আপনি/ *Apani* (you) is dropped.
Now since most of the cases the pronouns refer to subject or object and obviously it demands detection of the subject and/or object drops. Otherwise, pronoun resolution system will wrongly detect the referent of the pronoun. Subject, object drops are frequent in Bengali. Our experimental result shows that 13.83% sentences have the subject/object drop in a small Bengali corpus comprising of 8,455 sentences.

6.2.1 Related work

Most of the studies on pro-drop are explored in highly pro-drop languages like Japanese ([30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41]), Chinese ([42], [43], [44], [45]), Italian [46] and some works in other languages like English ([47], [48]), etc. The earlier approaches to drop resolution are based on manually created heuristic rules. However, the recent research trend of drop resolution has shifted from such rule-based approaches to machine learning based approaches ([42], [49] , [50]).

6.2.2 Proposed Method

Many language tree-banks mark drops as to facilitate the subsequent research on automatic detection of drops but we do not have any such resource for Bengali. For example, the Penn Tree bank [51] marked empty categories and this practice is followed in preparing Chinese Tree bank [52], Hindi Dependency Tree bank [53]. In case of Bengali, ICON 2009 [54] provided a Bengali Dependency Tree bank where only dropped verbs and conjunctions were marked.

In this experiment, both rule based approach and machine learning approaches have been applied to identify the subject/object drops in Bengali sentences. The rule based approach makes use of two simple rules. The intention of configuring the rule-based system is two-fold: (i) to investigate the capability of a simple rule based system for subject/object drop detection and (ii) to use this rule-based system in annotating data and thereby reducing the manual effort for preparing the annotated data set. Two statistical classifiers (namely, CRF-based and SVM-based) are then trained on the dataset generated by the semi-automatic way. Details about these systems are given in section 6.2.2.1 and section 6.2.2.2 respectively.

6.2.2.1 Rule Based Drop Detection

Our rule-based approach consists of just two linguist rules which are given below.
Rule 1: This rule checks the presence or absence of a subject in a given sentence. The absence of a subject can lead us to recognize it as a subject drop. A typical example of a Bengali sentence with subject drop is:

- ফুটবল খেলছেন।/ *phuTabala khelaChena* (playing football).

In this example, the subject তিনি/ *tini* (he) is dropped.

Additional check: This step performs an additional check to detect the presence or absence of proper noun and pronoun in a given sentence. Absence of these entities helps to confirm the presence of a subject drop. For example, if *Rule 1* is satisfied, i.e. if a subject drop is present in the sentence, then we further check whether the sentence contains no proper-noun and pronoun. This check confirms that there is a subject drop. The main reason behind *additional check* is to overcome the errors made by the dependency parser during subject tagging. In Bengali, the common tendency is to use a pronoun or proper noun in the subject position of a sentence (also, common nouns may serve the same role but presence of a common noun is not checked in our method because common nouns often appear in prepositional phrases, which may not serve as direct subject or object of the main verb).

Rule 2: The second rule contributes to object drop detection in a sentence. Let $count_1$ be the number of objects present in the sentence (as identified by the dependency parser [54]) and $count_2$ be the number of objects that the root verb of the sentence can take. $count_2$ is obtained from the verb category dictionary. If $count_1$ is less than $count_2$ we can say that there is an object drop in the sentence. A typical example of a Bengali sentence with object drop is:

- তিনি খেলছেন। / *tini khelaChena* (he is playing).

In this example, the main verb খেলছেন/*khelaChana* (is playing) is mono-transitive(i.e. the verb takes one direct object) by nature. But, the object, i.e. ফুটবল/ *phuTabala* is dropped in the sentence.

These two rules are used in conjunction to detect drops. The first rule attempts to capture subject drop, whereas the second rule tries to detect the object drop in a sentence.

6.2.2.2 CRF and SVM-based Drop Detection

Bengali is a free word order language and therefore, a dropped subject or object cannot be positioned in fixed places in a sentence. Thus, we assume the problem of identification of dropped subject or object on a sentence level. Therefore, each sentence in the training data is tagged as either "Drop" or "No-Drop". This results in a 2-class classification problem at the sentence level. In this experiment, we have used Conditional Random Field (CRF) [55] and Support Vector Machines (SVM) [56] as classifiers for detection of drop in a sentence. Both of the classifiers use the same features for the task. Some of the features are directly taken from the language dependent rules discussed in section 6.2.2.1 and other features are adopted from previous works by other researchers (as described in section 6.2.1). The complete list of features with their meanings are listed as follows:

- *Subject information:* This feature value is true if the subject is present in the sentence, otherwise false.
- *Proper noun information:* This feature value is true if the proper noun is present in the sentence, otherwise false.
- *Pronoun information:* This feature is true if a pronoun is present in the sentence, otherwise false.
- *Object counts:* This is also a binary feature. If $count_1$ is less than $count_2$ then this feature is true, otherwise false.
- *Current Word:* The current word of each sentence is used as a feature.
- *POS and Chunk Information:* We have used POS and Chunk information of a particular word as a feature.
- *First Word and its POS:* The first word with its POS information of each sentence is used as a feature.

- *Number of noun phrases:* The total number of noun phrases present in each sentence is also used as a feature.

For implementing the CRF-based classifier, we have used CRF++-0.58 [57]. For fast training, LBFGS, a quasi-newton algorithm [58] is used. For SVM-based classifier, we have used YamCha 0.33 [59] tool kit for detecting classes in documents. The polynomial kernel function used and the SVM parameter list is set to "*-t1 -d2 -c1*", which means the 2^{nd} degree of polynomial kernel and 1 slack variable is used. We use the TinySVM-0.09 classifier [60].

6.2.3 Experimental Setup

6.2.3.1 Creation of the Annotated Dataset

The Bengali corpus [61] is a collection of popular Bengali novel চোখের বালি/ *chAekhara bAila* and short story প্রায়শ্চিৎত/ *prAyaschitta* written by the Nobel Laureate, Rabindranath Tagore. The Corpus contained 8,455 sentences consisting of 1,11,052 tokens. Both of the rule based approaches and statistical approaches need the following pre-processing steps.

Step-1: The first step is POS tagging. The Stanford POS tagger [62] has been retrained for Bengali language. The tagger is trained on about 1,30,000 tagged words and results in 94% accuracy while testing on 2,000 words.

Step-2: The second step includes chunking of the data. We have used YamCha [59], a support vector machine based chunker. The chunker is trained with 2,685 sentences.

Step-3: In the third step, the data is parsed. A statistical dependency parser developed by Das et. al. [63] is used to get the parsed information [64]. A list of 2,600 sentences is divided into 5 sets to facilitate a 5-fold cross validation of the parser. Table 6.1 shows the parser accuracy (averaged over five folds) on the subject, object and root verb detection.

Step-4: After getting the main verb information from the parsed result, the respective verb category, (i.e., mono-transitive, intransitive and di-transitive) is tagged with the help of the verb-category dictionary. Complete statistics of the verb-category dictionary and its coverage are given in Table 6.2.

Step-5: The final step includes tagging the data with the presence and absence of drop information. To reduce manual effort, we have executed our rule based algorithm (as described in section 6.2.2.1). Then, these tags have been checked manually and corrected by a linguist.

Table 6.1: Performance evaluation of parser

	Recall	Precision	F1
Root Verb	0.9082	0.9000	0.9041
Subject	0.6592	0.5054	0.5716
Object	0.5891	0.4529	0.5111

Table 6.2: Verb-category with number of objects they take.

		No. of verbs	
Category	#Object	Dictionary	# Instances in Corpus
Intransitive	0	145	3,464
Mono-transitive	1	33	3,899
Di-transitive	2	703	1,011
Total		881	8,374

In the Table 6.2 the numbers under “Dictionary” indicates the number of root verbs found in the dictionary and the numbers under “Corpus” indicates the number of instances (all morphological variants of the root verbs) in the corpus.

6.2.3.2 Result

The rule-based classification is attempted on the whole corpus (containing 8,455 sentences). The classification results are presented in the first row of Table 6.3. Results are presented in terms of standard metrics namely, precision, recall and F1-measure. The last column shows sentence-level classification results. In the data set, there are 1,170 sentences with subject/object drop and the rest are without any drop. It is seen that overall the rule-based classification can correctly detect 76% of the sentences. However, this approach shows weaker performance in detecting drops (when there is one) and performs better in detecting no-drops (when there is actually no drop). The results obtained by applying the rule-based method is manually checked and corrected, wherever required. The corrected data set is divided into 5 folds (each fold containing 1,691 sentences) to implement a 5-fold cross validation of the CRF- and SVM-based classifiers. The performances of CRF- and SVM-based classifiers are presented in the second and third row of Table 6.3. In both cases improvements over performance of the rule-based method are statistically significant (p-value < 0.01 in a two-tailed paired t-test). However, the results given by CRF- and SVM- based classifiers are comparable and the difference is not statistically significant (p-value > 0.05 in a two tailed paired t-test). In absolute terms, the SVM based classifier shows slightly better performance in detecting actual drops than that of CRF-based one. Integration of the classifier results has not been attempted here; however, assembling of these results may produce better performance.

Table 6.3: Performance evaluation of the rule-based approach followed by CRF and SVM

	Drop			No-Drop			Overall			
Method	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	% Correct
Rule-based	.3146	.6111	.4153	.9264	.7864	.8506	.7619	.7621	.7620	76%
CRF	.5378	.2615	.3519	.8900	.9639	.9255	.8663	.8661	.8662	87%
SVM	.4596	.4769	.4681	.9151	.9100	.9125	.8497	.8495	.8496	85%

6.2.3.3 Error Analysis on Drop Detection

The result shows that sentences with drop are detected with less than 50% accurately. The main reason behind this low accuracy is the parser's low accuracy in detecting subject and objects (parser result is in Table 6.1). This in turn shows that the features taken from the parsed data are playing a vital role and mistakes made by the parser are hardly corrected by the other features used in classification. This observation conforms to the findings of the previous studies. For example, Yang and Xue [65] showed that for detecting empty category detection an F-score of 63.2% was achieved using automatic parsed data given by a parser working on accuracy of 80.3% (F1-score). However, when they used a gold standard parsed data they could improve the detection result by 25.8% (F1-score). In our study, the errors introduced by the POS tagger and chunker also result in some drop detection errors as these two tools normally operate at around 85% accuracy.

6.3 One-Expression and One-anaphora in Bengali

One-anaphora is one of the important anaphoric categories and it is present in all languages. In this section, we will study the corpus based analysis of one-expression and one-anaphora. The one-expressions also play important role in many other areas of NLP, for instance, question answering, machine translation, etc. For instance, consider the following sentences in Bengali:

- এক সময় সেখানে এক রাজা ছিলেন / *eka samay sekhAne eka rAjA Chilena* (Once upon a time, there was a king)

There are two one expressions (both are এক/*eka*) in the above sentence. While translating this sentence into English, the first one expression is translated to “once” and the second one expression is translated to “a”. There are instances when the same one-expression (e.g. এক/*eka*) is used in an inflected form and is translated to the number “one”. For example,

- বাজারে একটাও লোক নাই/ *bAjAre ekaTAo loka nAi* (There is no one in the market)

In this sentence, the one-expression, একটাও/*ekaTAo* (an inflected form of এক/*eka*) is translated to “one”. Sometimes, the one-expression is not translated at all. For example, consider this sentence,

- রাম ও শ্যাম কে এক করে দেখা ঠিক নয়/ *rAma o shyAma ke eka kare dekhA Thika nay* (It is not justified to treat Ram and Shyam equally)

In this sentence the one-expression এক/*eka*(one) has not been translated at all.

- আমার কাছে একজন কে পাঠাও ... / *AmAra kAChe ekajana ke pAThAo ...* (Send someone to me ...)

In this sentence the one একজন/*ekajana*(someone) is anaphoric and it refers to a person.

The above discussion shows that the same one expression behaves differently in different context. So for one-anaphora, at first we must have to identify the anaphoric-one. Here we not only concentrate on one-anaphora identification, but also consider the overall behaviour of one-expression. The distinct contributions of this work refer to (i) an exhaustive study of Bengali one-expressions (from a 177 million-word corpus) and their classification, (ii) preparation of two annotated data sets (details are in section 6.3.3): the first one containing 1806 sentences consisting of 2006 instances of one-expressions and the second one containing 296 sentences consisting of 300 instances of one-expressions. Each one-expression in these data sets is tagged with their respective class, (iii) study of the features contributing significantly for classification of one expression and then developing a CRF-based classifier for automatic classification of the one-expressions. One (the bigger one) of the annotated datasets is used to train the CRF-based classifier which is tested on the second data set.

6.3.1 Previous Studies

The computational analysis of one-expression in Bengali has not been explored earlier. Even there is hardly any linguistic study on classification of one-expressions in Bengali. Computational hardship refers to unavailability of annotated data sets (marking one-expressions in sentences and then tagging them with their respective classes). However, statistics show that in Indic languages like Bengali, one-expressions are used often. A study on 177-million-word FIRE Bengali Corpus [66] shows that about 1.34-million words refer to one expression. Obviously, they demand additional processing effort for NLP applications. This finding conforms to the observation of Hwee Tou Ng et. al. [67] who showed the statistical measure of the word “one” in the 100-million word British National Corpus (BNC) and claimed that one cannot just ignore “one” in any NLP application.

In case of English, one-expressions have been studied while dealing with one-anaphora ([68], [69], [70], [71]). Halliday and Hasan [68], Dahl [70] and Luperfoy [71] identify major criteria that distinguish the non-anaphoric uses. Hwee Tou Ng et. al. [67] classified the uses of one into six classes: (i) Numeric-one (I have only one blue T-shirt), (ii) Partitive-one (A special exhibition of books for the child forms one of the centrepieces), (iii) Anaphoric-one (Would you like this book? Yes, I would like that one), (iv) Generic-one (One must think a little deeper to find out the underlying social roots of the problem), (v) Idiomatic-one (It would be perfect to have a loved one accompany me on the whole trip), and (vi) Unclassifiable-one (Cursed be one who curses you). Out of these classes, they concentrate on the anaphoric class and used a machine learning approach to the identification and resolution of one anaphora. In our study, we follow the classification scheme of Hwee Tou Ng et. al. [67] for classifying the Bengali one-expressions with some extension.

6.3.2 Properties of one-expressions in Bengali

One expression in Bengali is more complex compared to English like languages. Since Bengali is the highly morphologically rich language, most of the words are highly inflected. In our experiment, we have identified twenty one commonly used such forms of one-expressions [এক/*eka* (no inflection), একটা/*ekaTA* (ek with *-ta* classifier), একটি/*ekaTi* (ek with *-ti* classifier), একটাই/*ekaTAi* (ek with *-tai* inflection), একটার/*ekaTAra* (ek with *-tar* inflection), ...].

The use of one-expression is quite frequent in Bengali. We investigated the frequency of one-expressions in the 177-million-word FIRE Bengali Corpus [66] and found that about 0.76% words (about 1.34 million words) in the corpus are one-expressions. This count includes all morphological variations of one-expressions. The frequency of one-expression clearly shows the dominant presence of one-expressions in Bengali. As a reference one may note that the words না/*nA* (no) and করে/*kare* (do) are the two most frequent words in FIRE corpus and their occurrence frequencies are 0.66% and 0.60%, respectively.

Classification of one-expressions is based on the instances found in the FIRE corpus. We follow the classification scheme of Hwee Tou Ng et. al. [67] with one exception. Instead of six classes we found seven dominant classes among the Bengali one-expressions. The Equality class (explained next) which is not relevant for English but found to be quite in use for Bengali. The seven classes are explained as follows:

Idiomatic one (IDO)

In Bengali it acts like a particle and generally associated with definite or indefinite singularity of any entity. Functionally, it is very similar to the indefinite/definite article “a/an, the” in English.

Example: একদা অযোদ্ধায় [এক] রাজা ছিল/ *ekadA ayoddhAy [eka] rAjA Chila*(Once upon a time there was [a] king at Ajyodha).

Numeric one (NUM)

It indicates the numeric (cardinal) value “one”.

Example: আমার কাছে মাত্র [এক] টাকা আছে/ *AmAra kAeCha mAtra [eka] TAkA AChe* (I have only [one] rupee).

Partitive one (PAT)

Selects an individual from a group of object.

Example: কোনও [এক] ঝুপড়িতে রান্নার সময়েই ওই আগুন লাগে বলে সন্দেহ করা হচ্ছে/ *[konao eka] jhuparite rAnnAra samayei oi Aguna lAge bale sandeha karA hachChe* (It is suspected that the fire broke out in any [one] of the huts from cooking oven)

Anaphoric one (ANA)

The one having a referent.

Example: ওর দুটো ওয়াকম্যন আছে , [একটা] আমি নিয়ে নেবো/ *ora duTo oyAkamyana AChe , [ekaTA]*

Ami niye nebo (He has two walkmans, I will take [one]).

Equality one (EQU)

Use of this one is for equality among two or more entities.

Example: সন্ত্রাসবাদীদের সঙ্গে গোটা ইসলামিক দুনিয়াকে [এক] করে দেখা ঠিক নয়/ *santrAsabAdIdera sa Nge goTA isalAmika duniyAke [eka] kare dekhA Thika nay* (It is not fair to see terrorists [together with] the whole Islamic world.)

Note: One interesting property in Bengali is the frequent use of the word একই/*ekai*(same) whose root form is এক/*eka*(one). But the expression একই/*ekai* do not come under one-expression.

Generic one (GEN)

A pronominal use that refers to a generic entity.

Example: প্রাথমিক ভাবে পুলিশের অনুমান, সভায় হাজির কেউ **[এক]** জন বোমাটি সঙ্গে নিয়ে এসেছিল/ *(prAthamika bhAbe pulishera anumAna, sabhAy hAjira keu [eka] jana bomATi sa Nge niye eseChila* (Primarily the police suspects that some [one] attending the meeting brought the bomb.)

Other one (OTH)

The one-expression other than above six classes.

Example: [এক] কথায়, রাজনীতির ঘূর্ণাবতে পড়িয়া বাংলা আজ নানা দিকেই পর্যুদস্ত *//[eka kathAy], rAjanItira ghUrNAbate pariyA bAMlA Aja nAnA dikei paryudasta* ([In brief], Bengal, in many aspects, is now in a disastrous condition due to its political turmoil.)

6.3.3 Preparation of annotated data

From the FIRE corpus, we randomly selected 1,806 sentences containing 2006 one-expression and manually annotated these with one of the seven classes described above. The distribution of each class in the annotated corpus is shown in the Table 6.4. We call this annotated dataset T_r as this has been used to train a CRF-based classifier as explained in the next section.

It is noted that this distribution of one-expressions differs from that of other languages. For example, the experiment was conducted by Hwee Tou Ng et. al [67] found Numeric class (46.9%) as the most frequent one followed by Partitive (25.3%). Idiomatic (1.6%) was seen to be very less frequent in their data set of 1,577 one-expressions randomly selected from the BNC corpus.

Table 6.4: T_r : Distribution of one-expression in the annotated data set

Class	Frequency	Percentage %
Idiomatic	544	27.12
Partitive	415	20.69
Numeric	362	18.05
Generic	266	13.26
Equality	114	5.68
Anaphoric	98	4.88
Other	207	10.32
Total	2006	100

6.3.4 Automatic Classification of one-expressions

We configured a CRF-based classifier for automatic classification of Bengali one-expressions. In our experiment, we have configured a Java-based open source package known as MAchine Learning for LanguagE Toolkit (MALLET)[1] . A set of seven features that contribute significantly for classifying the one-expressions are identified with the help of linguists. Description of these seven features is given below:

POS tag of One (W_0): We have considered the POS of the one-expression as a feature. In our experiment we have found the POS of one-expression is either QC (cardinal) or NN (common noun). For POS tagging we have used the Bengali POS tagger as discussed in section 3.2

Inflections (classifier) of One: The inflection (classifier) of a one-expression is considered as a feature. We have twenty-one such inflections (and classifiers) *-ta, -ti, -tai, -tir, -tite*, ...

Previous word (W_{-1}) of One: The immediate previous word of the one-expression.

[1] http://mallet.cs.umass.edu/sequences.php

Next word (W_{+1}) of One: The immediate next word of the one-expression.

Sentence starts with One: Whether the one is the starting word of the sentence.

Sentence ends with One: Whether the one is the ending word of the sentence.

Measuring unit followed by One: Whether the next word of one is measuring unit (like thousand, kilogram,etc.).

6.3.5 Training Data

The annotated dataset W_r is used for training the CRF. The sentences in W_r are POS tagged and the one expression is tagged with their respective class labels. The annotated dataset is presented in a column format as shown in Table 6.5 and Table 6.6 shows the detailed description of the data format. The CRF-based classifier uses maximum likelihood for training, for feature expectations it uses the forward backward algorithm and a Gaussian prior for parameter optimization.

Table 6.5: Training data format

....	...	....	....	...
txt1.txt	0	১১	QC	o
txt1.txt	1	সেপ্টম্বরের	NN	o
txt1.txt	2	এক	QC	NUM
txt1.txt	3	সপ্তাহ	NN	o
txt1.txt	4	আগেই	NST	o
txt1.txt	5	মার্কিন	NN	o
txt1.txt	6	প্রশাসন	NN	o
....	...	....	....	...

Table 6.6: Description of training data

Column id	Type	Description
1	Document Id	Contains the file name
2	Word number	Word index in the sentence
3	Word	Word itself
4	POS	POS of the word
5	Classification	Classification tag

6.3.6 Evaluation

The configured classifier has been evaluated by the publicly available ICON 2011 data set [14] which was prepared primarily for Bengali anaphora resolution. This data set consists of nine text pieces and we have extended this data set by adding four more texts. This combined data set (T_e) has been used for the evaluation of the anaphora resolution systems presented in Chapter 5. The choice of this data set is somewhat intentional as this data set has been annotated for anaphora resolution. As one-anaphor is one of the one-expression classes, annotation with one expression information would help subsequent research on resolution of one-anaphora. The data in T_e is presented in same format as shown in Table 6.5. Table 6.7 shows the coverage of T_e in terms of number of text pieces, words and one-expressions.

Table 6.7: T_e : Coverage of test data

Data	Test data
#texts	13
#words	27,454
#one-expressions	300

Table 6.8: Result of one-expression classification

Class label	#Intances	#Correct classification	#Incorrect classification	Recall	Precision	F1-score
IDO	103	85	25	0.83	0.77	0.80
NUM	84	77	37	0.92	0.68	0.78
OTH	43	29	3	0.67	0.91	0.77
PAT	32	12	0	0.38	1.00	0.55
GEN	17	7	1	0.41	0.88	0.56
ANA	16	13	8	0.81	0.62	0.70
EQU	5	3	0	0.60	1.00	0.75
Total	300	226	74	0.75	0.75	0.75

Table 6.8 gives the results for the one-expression classification for each of the seven classes. The average accuracy of one-expression classification is about 75% whereas the accuracy of Idiomatic (80%), Numeric (78%) and Partitive (77%) are relatively better.

As far as recall and precisions are concerned, the NUM class shows the highest recall and the PAT class shows the highest precision. The most dominant class, i.e. IDO shows the highest F1-score.

6.3.7 Error Analysis on one expression classification

Most of the errors occur due to inter-class confusion. Table 6.9 shows the confusion matrix that shows that IDO and NUM are two classes which create major confusions. For all other classes, a dominant tendency is to be confused with either IDO or NUM classes. This is because, some features (classifier/inflections, e.g., -ta/-ti; POS tags QC/cardinal, etc.) are strongly favourable for Idiomatic and Numeric classes. Many instances of PAT class are also confused, but such confusions are spread over three different classes, confusion with the ANA is the most significant. This is because the features for Partitive (PAT) class are much closed to the features of Anaphoric (ANA) class. In fact, some instances of Partitive class is a special kind of anaphoric one-expressions.

Table 6.9: Confusion Matrix for one-expression classification

	IDO	NUM	PAT	ANA	EQU	GEN	OTH
IDO	x	17	0	0	0	0	1
NUM	6	x	0	0	0	0	1
PAT	5	6	x	8	0	1	0
ANA	1	1	0	x	0	0	1
EQU	2	0	0	0	x	0	0
GEN	6	4	0	0	0	x	0
OTH	5	9	0	0	0	0	x

6.4 Role of Verbs

In Anaphora resolution it not only involves the syntactic information, but also involves many other factors, like semantic knowledge, real world knowledge, discourse knowledge, pragmatic knowledge, etc. Some of these issues for Bengali has been identified in early linguistic work [24]. Verbs play the very important role in sentence construction. It establishes the relations among the words in a sentence and hence comprehension. It is well accepted that some crucial aspects of a sentence are determined by the semantic and syntactic attributes of the verb that appears in the sentence. A verb has some expectations for its arguments whereas noun has compatibility for going with a specific verb. This property has an important role in anaphora resolution. Consider the following examples:

- টেবিল থেকে মিষ্টিটা তোল, ওটা আমি খাব / *Tebila theke miShTiTA tola, oTA Ami khAba* (take the sweet from the table, I will eat it)
- টেবিল থেকে মিষ্টিটা তোল, ওটা আমি ধোব / *Tebila theke miShTiTA tola, oTA Ami dhoba* (take the sweet from the table, I will wash it)

Now when our PEA approach (Chapter 5) resolves the anaphora ওটা/ *oTA* (it) then algorithm will find the possible antecedents as টেবিল/ *Tebila* (table) or মিষ্টিটা/ *miShTiTA* (sweet), and finally, chooses the মিষ্টিটা/ *miShTiTA* (sweet) in both the cases, but which is not correct for the second sentence. But to resolve ওটা/ *oTA* (it) correctly in these cases we have to consider the role of verb. In the first sentence, the verb খাব/ *khAba* (eat) takes only the arguments those are eatable things and hence it must be মিষ্টিটা/ *miShTiTA* (sweet) and hence the correct resolution is মিষ্টিটা/ *miShTiTA* (sweet). Similarly in the second sentence the verb ধোব/ *dhoba* (wash) takes only the arguments those are washable things and hence it must be টেবিল/ *Tebila* (table).

6.4.1 Verb compatibility with respect to nouns

Verbs cannot take any number of arguments; actually the number of arguments depends on the transitive information of the verb. Also, verbs cannot take every noun as an argument. Consider the following examples:

- পাখি আকাশে উড়ছে / *pAkhi AkAshe urChe* (bird is flying in the sky)

- মাছ আকাশে উড়ছে / *mACha AkAshe urChe* (fish is flying in the sky)

In the above examples, though both are grammatically correct but the first one is acceptable and the second example is not acceptable. The main difference between the sentences is the compatibility of noun with respect to the verb *fly*. Therefore, for anaphora resolution one challenge is to find out the correct argument with respect to a given verb. For this purpose, we have used the system developed by CDAC Kolkata [72] “Valency Analyzer of Verb Arguments for Bangla” for the noun compatibility with respect to verb. The system classifies the verbs based on their transitivity and the ontological class hierarchy of objects ([73], [74]).

6.4.2 Experiment and result

To check the importance of the feature (compatibility of noun with respect to verb) in anaphora resolution the following experiment is carried out. Note that, for this experiment specific type of data is required, i.e. instances having two or more possible antecedents and can be resolved by checking the valid verb argument. But a considerable number of such types of instances are not available in our earlier data set [14]. Hence we have collected 100 such instances from the corpus in [66]. Note that the instances are individual sentences not in a discourse. This data set is represented in column format as described in Chapter 5, section 5.3. Our PEA algorithm (as explained in Chapter 5) is applied on this data set. The system originally resolves 43 cases correctly. Next, when the verb compatibility is considered then it could resolve 53 cases correctly and hence the improvement is noticeable. An important observation is that the PEA approach, in general, gives better result as was shown on the ICON2011 [14] data set. To investigate about the low accuracy of PEA on this specially designed data set, the error analysis is done to find out the nature of errors made by this system. Here three major shortcomings are identified as stated below.

(i) In the PAR system used some high confidence rules (section 5.7), but here such rules are not applicable on this data.

(ii) For this experiment the data set is specially generated, i.e. in all cases there exists an ambiguity. In case of conflict, the PAR system used some syntactic agreement to resolve the ambiguities, and if still not resolved, then use a heuristic approach to choose the most recent one. On the other hand, when the ICON2011 [14] data used in the PAR system, there is a less number of ambiguities on that data set.

(iii) Here valid verb argument is used additionally alone with PAR system. But the data show that only this information is not sufficient to resolve the ambiguity, instead it require other knowledges like discourse knowledge, real world knowledge, etc.

Consider the following examples:

- ইটিকুইরা জলপ্রপাত দেখে আবার ব্রাসিলিয়া ফিরে এসে সেখান থেকে দেশে ফিরব / *iTikuirA jalaprapAta dekhe AbAra brAsili A phire ese sekhAna theke deshe phiraba* (I will return to Brasilia after visiting Etiquira falls and from there I will return to my counter)

- পেন-ড্রাইভটা কম্পুটার থেকে খুলে রাখ, ওতে আর ফাঁকা জায়গা নাই / *pena-DrAibhaTA kampuTAra theke khule rAkha , ote Ara phA.NkA jAygA nAi* (Take out the pen-drive from the computer, there is no space in it)

- ... সব থেকে অবাক করা ওর পাখনা, যেন অঙ্গ-প্রত্যঙ্গর মতই ওগুলো দেহ থেকে বার হয়ে আছে / *... saba theke abAka karA ora pAkhan , yena a Nga-pratya Ngara matai ogulo deha theke bAra ha e AChe* (... the most wonderful things are its fins which have come out of its body like the other parts)

- বাইক থেকে ব্যাগটা তুলে রাখ, ওটাতে বাজার আছে / *bAika theke byAgaTA tule rAkha , oTAte bAjAra AChe* (take the bag from the bike, it contains vegetable)

- ... মাংস আর কুকার সঙ্গেই এনেছে, বলে আজই এটা বানিয়ে খাব / *mAMsa Ara kukAra sa Ngei eneChe , bale Ajai eTA bAniye khAba* (... has brought meat and cooker with him and says that he will cook it to eat it today itself)

When our PEA system tried to resolve the anaphora shown above by underlines, most of the cases it encounters ambiguity problem and chooses the most recent one and hence it solved 43 cases out of 100 correctly. Particularly in the above examples, it resolves 1^{st} and the 4^{th} one correctly and resolves the 2^{nd} and 5^{th} one incorrectly. Where, in the 3^{rd} case the system leaves it unresolved because of the mismatch of number agreement with anaphora and the antecedent. When the system includes the verb agreement feature, then it resolves 1^{st}, 4^{th} and 5^{th} correctly. Note that in PEA system the 5^{th} one was resolved incorrectly because of the heuristic approach (choose most recent one), but when the system includes the verb argument feature it resolves correctly. Also note that in the 2^{nd} example, it also fails to resolve because only the verb argument is not sufficient to resolve it but the real world and discourse knowledge are required. Similarly, in the 3^{rd} example, it also requires the real world knowledge to resolve it correctly.

6.5 Summary

This chapter mainly discusses three advanced research issues with respect to the Bengali language. The issues are not only important for anaphora resolution, but also important for several other NLP applications. Since there is no significant computational work on these issues, our work provides a number of important insights in context of computational linguistics of Bengali. At first, a corpus based analysis is provided that investigates quantitative measures of pro-drop and one-expressions in the text and the importance of such measures is explained. Necessary tools and resources for conducting this research are mentioned. Development of computational approaches is also presented. Finally, influence of such issues on development of robust anaphora resolution system is considered. In summary, this chapter enlightens some future research areas in the domain of discourse analysis in Bengali.

Chapter 7

Conclusion and future work

The motivation behind the present thesis was towards the advancement of research on the pronominal anaphora resolution in Bengali, a sub-problem in the NLP domain. In the global scenario when the researchers are trying to shift on statistical (machine learning) methods and trying to use the web resources, research in Indian languages is still starving on unavailability of basic resources. Hence, the major goals of thesis were set in three fronts, the first one was the basic resource building, the second one was the methodology and the third one was to address the advanced issues in anaphora resolution for Bengali.

7.1 Goal

The thesis had set the following three major goals.

7.1.1 Resource Building

Here we defined the resources from two perspectives, i.e. the first one is the technical resource, the basic NLP tools and the second one is the language resource, the necessary linguistic features and properties, relevant for anaphora resolution.

- *Basic NLP tools :* As said earlier the anaphora resolution can't be done on raw text, a lot of pre-processing is required. But such pre-processing NLP tools like POS-tagger, NER, Morphological analyzer, etc. are not available. Though in literature some of the systems are reported, but they are either not available in ready-to-use mode or it is difficult to re-produce. Hence, our first target was to make availability of such tools.

- *Linguistic features and properties:* Since, the thesis mainly concentrates on pronominal anaphora resolution, so the pronouns are the most import element in this task. As we know the Bengali is a highly inflected language, a large set of pronouns is there with inflected form. The inflections play a crucial role in defining syntactic as well as functional meaning. On the other hand, a lot of exceptions present in Bengali pronoun. So, another goal was the extensive study of Bengali pronouns to identify such properties and exceptions to incorporate such issues in anaphora resolution system.

7.1.2 Methodology of Anaphora Resolution

In designing the methodology we had set following goals.

- *Adaptation of existing system:* There are several off-the-shelf systems available for anaphora resolution. But most of these are for English language and hence it is difficult to use such systems directly for the language of other families like Indic languages. The main problem behind the adaptation of such systems is the language dependency. Our target was to find out such dependencies and change accordingly to fit the system for Bengali.

- *Development of a new methodology:* Though there are several approaches available in anaphora resolution, but most of these are designed for the English like languages. The main problems to use such algorithms are the underling pre-processing tools and the language constraints. The complex feature of pronoun is an additional problem for Bengali. Hence our target was to develop a suitable method for the Bengali language.

- *Incorporation of machine learning approach:* For the last decades, researchers are trying to apply the machine learning approach in several NLP tasks including anaphora resolution. We also planned to use machine learning techniques in our work.

7.1.3 Advanced issues in Anaphora Resolution

- Anaphora resolution is a complex problem, only grammatical properties are not sufficient to solve the anaphora, instead it requires semantic knowledge, pragmatic knowledge, real-world knowledge, etc. So far most of the work done in the Indic language is used the grammatical features in the algorithm. Also, most of the works are in pronominal anaphora, because most dominating types of anaphora are pronouns. Our target was to go beyond the current trend, i.e. wanted to use some advanced features in anaphora resolution.

7.2 Achievement

7.2.1 Achievement in Resource Building

We have successfully developed and configured some basic NLP tools for anaphora resolution and also done a corpus based study for Bengali pronouns.

- *Basic NLP tools (POS-tagger, NER, Morphological analyser) :* The Stanford POS-tagger has been configured successfully for Bengali language. For training the tagger ICON2011 [14] tagged data and LDC tagged data [75] for Bengali are used. A rule based NER system has been developed and a morphological analyser is developed based on the working principle of a finite automata (Chapter 3).

- *Linguistic features and properties:* The corpus based study on Bengali pronouns gives a statistical measure along with some other valuable information which are relevant for anaphora resolution (Chapter 2).

7.2.2 Achievement in Methodology

We have fulfilled our initially set goals in methodology as briefed below.

- *Adaptation of GuiTAR:* GuiTAR [5] is a well known off-the-shelf anaphora resolution system primarily designed for English. We have successfully configured the system for Bengali with necessary changes (Chapter 4).

- *Development of a new algorithm:* A rule based, pronoun emitting approach has been developed for anaphora resolution (Chapter 5). The concept of pronoun emitting is the new contribution in the community. The system is competitive enough compared to the other existing work in Bengali.

- *Incorporation of machine learning approach:* So far, we have overcome the problem with the basic tools for anaphora resolution. Still, it is difficult to developed an end-to-end anaphora resolution system exclusively following machine learning approaches because of the lack of annotated corpora. We have incorporated the machine learning approach for identifying the honorific information about a person and the information has been used in anaphora resolution. Also, Stanford POS-tagger is based on the machine learning approach, which has been reconfigured for Bengali. We have also explored the use of conditional random field (CRF) and support vector machines (SVM) based approach for analysis of pro-drops and one-expressions.

7.2.3 Advanced issues in Anaphora Resolution

- Under advance issues, we have focused on three different types of analysis namely, *pro-drop*, *one-expression* and effect the *verb argument* in anaphora resolution (Chapter 6). A corpus based study has been carried out for one-expression and it explored the frequency of one-anaphora in the text. The pro-drop analysis and a pro-drop resolution strategy have been developed. It is also identified that (Chapter 5, section 5.7.2) how sometimes pro-drops are misleading in anaphora resolution. Finally, an experiment was done on the role of verb on anaphora resolution.

7.3 Scope of future work

Based on our knowledge, this is one of the pioneering computational attempts in anaphora resolution in Bengali. The studies of this thesis enlighten several future research directions, some of them are listed below.

- This thesis reveals that the existing off-the-shelf anaphora resolution systems are not adequate for Indic languages; instead the rule based systems perform better in terms of accuracy. It also investigates the shortcomings of an off-the-shelf system over the rule based system. This brings out future issues in order to improve the accuracy further.

- So far, the major research in Bengali anaphora is related to pronouns only, because of its dominating nature in the language, but some other types of anaphora do exist; and our corpus based study has established this with evidences (Chapter 2). The thesis also highlights another type (i.e. one-anaphora) and it has quite important role in discourse analysis. To develop a sophisticated anaphora resolution system all such types of anaphora must be addressed.

- Most of the anaphora resolution approaches use the syntactic information to resolve the anaphora, but in many cases it also requires the semantic knowledge, pragmatic knowledge, real-world knowledge, etc. Here we have separately investigated one such issue, the verb argument with respect to the noun and shown its impact on anaphora resolution. But still several issues are unexplored.

- The thesis has also highlighted another important issue in NLP, i.e, the pro-drop. It has a deep impact on anaphora resolution, but still not explored properly.

- So far all the anaphora resolution systems in Bengali takes the input as processed data up to some extent, i.e. at least POS tags, NE tags and chunking information, etc. are required. Therefore, development of an end-to-end anaphora resolution system still remains a challenge. The main constraints behind development of an end-to-end system are highlighted in this thesis.

Publications out of this work

1 . **A. Senapati** and U. Garain, *“Anaphora resolution system for bengali by pronoun emitting approach,”* In the Proceedings of International Conference on Natural Language Processing (NLP Tool Contest), (ICON 2011), pp. 21-26, 2011.

2 . **A. Senapati** and U. Garain, *“Anaphora Resolution in Bangla Using Global Discourse Knowledge,”* In the Proceedings of International Conference of Asian Language Processing (IALP 2012), pp. 49-52, 2012.

3 . **A. Senapati** and U. Garain, *“Bangla Morphological Analyzer using Finite Automata,”* In the Processing of Forum of Information Retrieval Evaluation, ISI Kolkata :@ FIRE MET, (FIRE 2012), pp. 101-106, 2012.

4 . **A. Senapati** and U. Garain, *“GuiTAR-based Pronominal Anaphora Resolution in Bengali,”* In the Proceedings of the 51st Conference of the Association for Computational Linguistics (ACL 2013), pp. 126-130, 2013.

5 . **A. Senapati**, A. Das, and U. Garain, *“Named-Entity Recognition in Bengali,”* Proceedings of the Forum of Information Retrieval (FIRE 2013), pp. 143-147, 2013.

6 . **A. Senapati** and U. Garain, *“Detection of named entities for un-tagged data in Bengali,”* In the Proceedings of the International Workshop on Machine Learning and Text Analysis, 2013.

7 . **A. Senapati** and U. Garain, *“Role of verbs in anaphora resolution in Bengali,”* In the Processing of the International Conference of Linguistic Society of India (ICOLSI 2013), pp. 201-203, 2013.

8 . **A. Senapati** and U. Garain, *"One-expression classification in Bengali and its role in Bengali-English machine translation,"* In the Proceedings of International Conference of Asian Language Processing (IALP 2014), pp. 162-165, 2014.

9 . A. Das, **A. Senapati**, and U. Garain, *"Automatic detection of subject/object drops in Bengali,"* In the Proceedings of International Conference of Asian Language Processing (IALP 2014), pp. 91-94, 2014.

10 . **A. Senapati** and U. Garain, *"A Maximum Entropy Based Honorificity Identification for Bengali Pronominal Anaphora Resolution,"* In the Processing of the 15th Conference of Intelligent Text processing and Computational Linguistic (CICLing 2014), pp. 319-329, 2014.

11 . **A. Senapati**, and A. Das, and U. Garain, *"Pro-drop resolution in Bengali,"* In the Proceedings of International Conference on Natural Language Processing (Student Paper Competition), (ICON 2014), 2014.

12 . **A. Senapati** and U. Garain, *"A Computational Approach for Corpus Based Analysis of Reduplicated Words in Bengali,"* In the Processing of the 16th Conference of Intelligent Text processing and Computational Linguistic (CICLing 2015), pp. 456- 466, 2015.

Bibliography

[1] R. Gaizauskas and K. Humphreys, "Quantitative evaluation of coreference algorithms in an information extraction system," *In Corpus-based and Computational Approaches to Discourse Anaphora, pp. 145-169, John Benjamins*, 2000.

[2] T. Morton, "Coreference for nlp applications," *In the Proceedings of the 38th Conference of the Association for Computational Linguistics (ACL 2000)*, 2000.

[3] R. Watson, J. Preiss, and E. Briscoe, "The contribution of domainindependent robust pronominal anaphora resolution to open-domain questionanswering," *In the Proceedings of the International Symposium on Reference Resolution*, 2003.

[4] L. Alonso and M. Fuentes, "Integrating cohesion and coherence for text summarization," *In the Proceedings of the Student Session of European Chapter of the Association for Computational Linguistics (EACL 2003)*, 2003.

[5] M. Poesio and M. A. Kabadjov, "A general-purpose, off-the-shelf anaphora resolution module: Implementation and preliminary evaluation," *In Proceedings of the International Conference on Language Resources and Evaluation (LREC 2004), Lisbon, Portugal*, 2004.

[6] D. Byron and J. Tetreault, "A flexible architecture for reference resolution," *In the Proceedings of the European Chapter of the Association for Computational Linguistics (EACL 2003)*, 2003.

[7] S. Pradhan, A. Moschitti, and N. Xue, “Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes,” *In the Proceedings of the 16th Conference on Computational Natural Language Learning (CoNLL 2012): Shared Task*, 2012.

[8] R. Mitkov, “Anaphora resolution,” *Longman*, 2002.

[9] L. T. Group, “The lt-xml toolkit,” *Software available at https://www.ltg.ed.ac.uk/software/ltxml2/*, 2003.

[10] E. Charniak, “A maximum–inspired parser,” *In the Proceedings of the Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL 2000)*, 2000.

[11] M. Poesio, “The mate/gnome annotation scheme for anaphora deixis, revisited,” *In the Proceedings of the SIGdial Workshop on Discourse and Dialogue*, 2004.

[12] R. Vieira and M. Poesio, “An empirically-based system for processing definite descriptions,” *Computational Linguistics, vol. 26, no. 4, pp. 539-593*, 2000.

[13] K. Bontcheva, M. Dimitrov, D. Maynard, V. Tablan, and H. Cunningham, “Shallow methods for named entity coreference resolution,” *In the Proceedings of the Workshop TALN*, 2002.

[14] L. Sobha, S. Bandyopadhyay, V. S. Ram, and A. Akilandeswari, “Nlp tool contest @icon2011 on anaphora resolution in indian languages,” *In the Proceedings of NLP Tool Contest, (ICON 2011)*, 2011.

[15] M. A. Kabadjov, “A comprehensive evaluation of anaphora resolution and discourse-new classification,” *Ph.D. thesis, Department of Computer Science University of Essex, pp. 33-57*, 2007.

[16] B. Baldwin, "Cogniac: High precision coreference with limited knowledge and linguistic resources," *In the Proceedings of the ACL Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts (ACL 97), pp. 38-45*, 1997.

[17] X. Luo, "On coreference resolution performance metrics," *In the Proceedings of the Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing, pp. 25-32*, 2005.

[18] M. Vilain, J. Burger, J. Aberdeen, D. Connolly, and H. Lynette, "A model-theoretic coreference scoring scheme," *In the Proceedings of the 6th Message Understanding Conference (MUC 6), pp. 45-52*, 1995.

[19] ——, "Algorithms for scoring coreference chains," *In the Proceedings of the 1st International Conference on Language Resources and Evaluation, pp. 563-566*, 1999.

[20] X. Luo, "On coreference resolution performance metrics," *In the Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2005.

[21] M. Recasens and E. Hovy, "Blanc: Implementing the rand index for coreference resolution," *Natural Language Engineering, 17(4), pp. 485-510*, 2011.

[22] A. Senapati and U. Garain, "Anaphora resolution system for bengali by pronoun enitting approach," *In the Proceedings of International Conference on Natural Language Processing (NLP Tool Contest), (ICON 2011), pp. 21-26)*, 2011.

[23] G. Sengupta, "Lexical anaphors and pronouns in selected southasian languages : A principled typology," *Ed. by B. C. Lust, K. Wali, J. W. Gair and K. V. Subbarao, pp. 277-332*, 2011.

[24] A. Majumdar, "Studies in the anaphoric relations in bengali," *Publisher: Subarnarekha, India*, 2000.

[25] U. K. Sikdar, A. Ekbal, S. Saha, O. Uryupina, and M. Poesio, “Adapting a state-of-the-art anaphora resolution system for resource-poor language,” *In the proceedings of International Joint Conference on Natural Language Processing, pp. 815-821*, 2013.

[26] S. Chatterji, A. Dhar, B. Barik, P. K. Moumita, S. Sarkar, and A. Basu, “Anaphora resolution for bengali, hindi, and tamil using randomtree algorithm in weka,” *In the Proceedings of the NLP Tool Contest, (ICON 2011)*, 2011.

[27] R. Mitkov and C. Barbu, “Evaluation tool for rule-based anaphora resolution methods,” *In the Proceedings of the 39th Conference of the Association for Computational Linguistics (ACL 2001) pp. 34-41*, 2001.

[28] R. Mitkov, “Outstanding issues in anaphora resolution,” *In Computational Linguistics and Intelligent Text Processing, LNCS, Volume 2004, pp. 110-125*, 2001.

[29] N. Chomsky, “Lectures on government and binding: The pisa lectures,” *Dordrecht: Foris Publications*, 1981.

[30] K. Seki, A. Fujii, and T. Ishikawa, “A probabilistic method for analyzing japanese anaphora integrating zero pronoun detection and resolution,” *In the Proceedings of the 19th International Conference on Computational Linguistics (COLING 2002), pp. 911-917.*, 2002.

[31] H. Isozaki and T. Hirao, “Japanese zero pronoun resolution based on ranking rules and machine learning,” *In the Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 184-191*, 2003.

[32] R. Iida, K. Inui, and Y. Matsumoto, “Zero-anaphora resolution by learning rich syntactic pattern features,” *ACM Transactions on Asian Language Information Processing (TALIP), Volume 6. Issue 4, Article 12*, 2007.

[33] H. Taira, S. Fujita, and M. Nagata, "A japanese predicate argument structure analysis using decision lists," *In the Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2008), pp. 523-532*, 2008.

[34] R. Sasano, D. Kawahara, and S. Kurohashi, "A fully-lexicalized probabilistic model for japanese zero anaphora resolution," *In the Proceedings of the 22nd International Conference on Computational Linguistics (COLING 2008), pp. 769-776.*, 2008.

[35] ——, "The effect of corpus size on case frame acquisition for discourse analysis," *In the Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics, pp. 521-529*, 2009.

[36] K. Imamura, K. Saito, and T. Izumi, "Discriminative approach to predicateargument structure analysis with zero-anaphora resolution," *In the Proceedings of the ACL-IJCNLP Conference Short Papers, pp. 85-88.*, 2009.

[37] Y. Watanabe, M. Asahara, and Y. Matsumoto, "A structured model for joint learning of argument roles and predicate senses," *In the Proceedings of the 48th Conference of the Association for Computational Linguistics (ACL 2010), pp. 98-102*, 2010.

[38] Y. Hayashibe, M. Komachi, and Y. Matsumoto, "Japanese predicate argument structure analysis exploiting argument position and type," *In the Proceedings of 5th International Joint Conference on Natural Language Processing, pp. 201-209.*, 2011.

[39] K. Yoshikawa, . Asahara, and Y. Matsumoto, "Jointly extracting japanese predicate-argument relation with markov logic," *In the Proceedings of 5th International Joint Conference on Natural Language Processing, pp. 1125-1133*, 2011.

[40] M. Hangyo, D. Kawahara, and S. Kurohashi, “Japanese zero reference resolution considering exophora and author/reader mentions,” *In the Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 924-934*, 2013.

[41] K. Yoshino, S. Mori, and T. Kawahara, “Predicate argument structure analysis using partially annotated corpora,” *In the Proceedings of the 6th International Joint Conference on Natural Language Processing, pp. 957-961*, 2013.

[42] S. Zhao and H. T. Ng, “Identification and resolution of chinese zero pronouns: A machine learning approach.” *In the Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 541-550.*, 2007.

[43] C. Chen and V. Ng, “Chinese zero pronoun resolution: Some recent advances,” *In the Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1360-1365*, 2013.

[44] N. Xue and Y. Yang, “Chasing the ghost: recovering empty categories in the chinese treebank,” *In the Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010), pp. 1382-1390*, 2010.

[45] ——, “Dependency-based empty category detection via phrase structure trees,” *In the Proceedings of NAACL-HLT, pp. 1051-1060*, 2013.

[46] R. Iida and M. Poesio, “A cross-lingual ilp solution to zero anaphora resolution,” *In the Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011), pp. 804-813*, 2011.

[47] M. Johnson, “A simple pattern-matching algorithm for recovering empty nodes and their antecedents,” *In the Proceedings of the 40th Conference of the Association for Computational Linguistics (ACL 2002)*, 2002.

[48] R. Campbell, “Using linguistic principles to recover empty categories,” *In the Proceedings of the 42th Conference of the Association for Computational Linguistics (ACL 2004)*, 2004.

[49] L. Rello, P. Suárez, and R. Mitkov, “A machine learning method for identifying non-referential impersonal sentences and zero pronouns in spanish,” *Procesamiento del Lenguaje Natural, 45, pp. 281–287*, 2010.

[50] A. Park and M. Hong, “Hybrid approach to zero subject resolution for multilingual mt - spanish-to-korean cases -,” *In the Proceedings of the 28th Pacific Asia Conference on Language, Information and Computation pp. 254–261*, 2014.

[51] M. Marcus, B. Santorini, and M. A. Marcinkiewicz, “Building a large annotated corpus of english: the penn treebank,” *Computational Linguistics - Special issue on using large corpora: II archive Volume 19 Issue 2, pp. 313-330*, 1993.

[52] N. Xue and M. Palmer, “Adding semantic roles to the chinese treebank,” *NLE, Vol:15(1), pp. 143-172*, 2009.

[53] C. Gsk, S. Husain, and P. Mannem, “Empty categories in hindi dependency treebank: analysis and recovery,” *In the Proceedings of 5th Linguistic Annotation Workshop, ACL, pp. 134-142*, 2011.

[54] S. Husain, “Dependency parsers for indian languages,” *In the Proceedings of ICON, 2009*, 2009.

[55] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, “Conditional random fields: Probabilistic models for segmenting and labeling sequence data,” *In the Proceedings of the International Conference on Machine Learning (ICML 2001), pp. 282-289*, 2001.

[56] V. N. Vapnik, “The nature of statistical learning theory,” *Springer-Verlag New York*, 1995.

[57] T. Kodu, “Crf++: Yet another crf toolkit,” *http://crfpp.googlecode.com/svn/trunk/doc/index.html*, 2005.

[58] G. Andrew and J. Gao, “Scalable training of l1-regularized log-linear models,” *In the Proceedings of the International Conference on Machine Learning (ICML 2007), pp. 33-40*, 2007.

[59] T. Kudu and Y. Matsumoto, “Use of support vector learning for chunk identification,” *In the Proceedings of CoNLL (CoNLL 2000)*, 2000.

[60] V. N. Vapnik, “The statisitcal learning theory (http://chasen.org/ taku/software/tinysvm/),” *Springer, 1998*, 1998.

[61] S. for Natural Language Technology Research, “http://www.rabindra-rachanabali.nltr.org/node/2.”

[62] K. Toutanova and C. D. Manning, “Enriching the knowledge sources used in a maximum entropy part-of-speech tagger,” *In the Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC 2000), pp. 63-70*, 2000.

[63] A. Das, A. Shee, and U. Garain, “Evaluation of two bengali dependency parsers,” *MTPIL Workshop, COLING, pp. 133-142*, 2012.

[64] U. Garain, “Resources: Bengali nlp resources,” *http://www.isical.ac.in/ utpal/resources.php.*

[65] Y. Yang and N. Xue, “Chasing the ghost: recovering empty categories in the chinese treebank,” *In the Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010), pp. 1382-1390*, 2010.

[66] “Forum for information retrieval evaluation data,” *http://fire.irsi.res.in/fire/static/data*, 2012.

[67] H. T. Ng, Y. Zhou, R. Dale, and M. Gardiner, "Machine learning approach to identification and resolution of one-anaphora," *IJCAI, pp. 1105-1110*, 2005.

[68] M. A. K. Halliday and R. Hasan, "Cohesion in english," *London: Longman*, 1976.

[69] B. Webber, "A formal approach to discourse anaphora," *New York: Garland Publishing*, 1979.

[70] D. A. Dahl, "The structure and function of one anaphora in english," *Ph.D. thesis, University of Minnesota*, 1985.

[71] S. Luperfoy, "Discourse pegs: A computational analysis of context-dependent referring expressions," *Ph.D. thesis, University of Texas at Austin*, 1991.

[72] S. Chandra and P. Bhattacharyya, "Valency analyzer of verb arguments for bangla," *In the Proceeding of 13th Oriental COCOSDA Workshop*, 2010.

[73] M. Jarmasz and S. Szpakowicz, "Roget thesaurus and semantics similarity," *In the Proceedings of the conference on Recent Advance in Natural Languages Processing, pp. 212-219*, 2003.

[74] B. Smith and C. Welty, "Ontology: Towards a new synthesis," *Formal Ontology in Information Systems, In Chris Welty and Barry Smith, eds. ACM Press*, 2001.

[75] K. Bali, M. Choudhury, and P. Biswas, "Indian language part-of-speech tagset: Bengali," *Linguistic Data Consortium, LDC2010T16*, 2010.

[76] Y. Huang, "Anaphora: A cross-linguistic approach," *Oxford: Oxford University Press*, 2000.

[77] P. Nand, "Resolving co-reference anaphora using semantic constraints," *Ph.D. Thesis, AUT University*, 2012.

[78] B. Butterworth, "Hesitation and semantic planning in speech," *Journal of Psycholinguistic Research, 4:75-81*, 1974.

[79] H. H. Clark and C. J. Sengul, "In search of referents for nouns and pronouns," *Memory and Cognition, 1(7):35-41*, 1979.

[80] P. C. Gordon, B. J. Grosz, and L. A. Gilliom, "Pronouns, names, and the centering of attention in discourse," *Cognitive Science, 17:311-348*, 1993.

[81] J. R. Hobbs, "Coherence and coreference," *Cognitive Science, 67:67-90*, 1979.

[82] P. N. Johnson-Laird, "Mental models: towards a cognitive science of language, inference, and consciousness," *Harvard University Press, Cambridge, MA, USA*, 1983.

[83] S. E. Brennan, M. W. Friedman, and C. J. Pollard, "A centering approach to pronouns," *In the Proceedings of the 25th annual meeting on Association for Computational Linguistics (ACL 87), pp. 155-162*, 1987.

[84] M. Kameyama, "A property-sharing constraint in centering," *In the Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics (ACL 86), pp. 200-206*, 1986.

[85] A. Kehler, "The effect of establishing coherence in ellipsis and anaphora resolution," *In the Proceedings of the 31st Conference of the Association for Computational Linguistics (ACL 93), pp. 62-69*, 1993.

[86] M. Mccord, "Anaphora resolution in slot grammar," *Computational Linguistics, 16:197-212*, 1990.

[87] R. Reichman-Adar, "Extended person-machine interface," *Artificial Intelligence, 22(2):157-218*, 1984.

[88] C. L. Sidner, "Towards a computational theory of definite anaphora comprehension in english discourse," *Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA*, 1979.

[89] R. Mitkov, “Anaphora resolution: the state of the art,” *Working paper. University of Wolverhampton, Wolverhampton*, 1999.

[90] M. Recasens, L. M‘arquez, E. Sapena, M. A. Marti, M. Taul , V. Hoste, M. Poesio, and Y. Versley, “Task 1: Coreference resolution in multiple languages,” *In the Proceedings of the 5th International Workshop on Semantic Evaluation (ACL 2010)*, 2010.

[91] M. Strube, “Never look back: An alternative to centering,” *In the Proceedings of the 36th Conference of the Association for Computational Linguistics (ACL 98), pp. 1251-1257*, 1998.

[92] J. Tetreault, “Analysis of syntax-based pronoun resolution methods,” *In the Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL 99), pp. 602-605*, 1999.

[93] Kennedy and Boguraev, “Anaphora for everyone: pronominal anaphora resolution without a parser,” *In the Proceedings of the 16th International Conference on Computational Linguistics (COLING 96), pp. 113-118*, 1996.

[94] R. Mitkov, “Robust pronoun resolution with limited knowledge,” *In the Proceedings of the 17th International Conference on Computational Linguistics (COLING 98/ACL 98), pp. 869-875*, 1998.

[95] M. Palomar, A. Ferrandez, L. Moreno, P. Martinez-Barco, J. Peral, M. Saiz-Noeda, and R. Mufioz, “An algorithm for anaphora resolution in spanish texts,” *Computational Linguistics, 27(4), 545-567*, 2001.

[96] ——, “A rule-based pronoun resolution system for french,” *In the Proceedings of the 4th Discourse Anaphora and Anaphora Resolution Colloquium (DAARC 2002)*, 2002.

[97] Y. Versley, S. P. Ponzetto, M. Poesio, V. Eidelman, A. Jern, J. Smith, X. Yang, and A. Moschitti, “Bart: A modular toolkit for coreference resolution,” *In the Proceedings of the Association for Computational Linguistics (ACL 08): HLT Demo Session (Companion Volume), pp. 9-12*, 2008.

[98] W. M. Soon, H. Ng, and C. Lim, “A machine learning approach to co-reference resolution of noun phrases,” *Computational Linguistics, 27 (4), 521-544*, 2001.

[99] D. G. Bobrow, “A question-answering system for high school algebra word problems,” *In the Proceedings of the Conference AFIPS, 26, 591-614*, 1964.

[100] T. Winograd, “Understanding natural language,” *New York: Academic Press/Edinburgh: Edinburgh University Press*, 1972.

[101] W. Woods, R. Kaplan, and B. Nash-Webber, “The lunar sciences natural language information system: final report,” *Report 2378. Cambridge, MA: Bolt Beranek and Newman*, 1972.

[102] J. R. Hobbs, “Pronoun resolution,” *Research Report 76-1. New York: Department of Computer Science, City University of New York*, 1976.

[103] ——, “Resolving pronoun references,” *Lingua 44:311-338*, 1978.

[104] M. Poesio, S. Ponzetto, and Y. Versley, “Computational models of anaphora resolution: A survey,” *Linguistic Issues in Language Technology.*, 2011.

[105] G. Hirst, “Anaphora in natural language understanding,” *Springer-Verlag Berlin*, 1981.

[106] W. Woods, “Progress in natural language understanding: An application to lunar geology,” *AFIPS Natl. Comput. Conj: Expo. 42, pp. 441-450*, 1973.

[107] E. Charniak, “Toward a model of children’s story comprehension,” *AI TR-266, Massachusetts Institute of Technology Artificial Intelligence Laboratory*, 1972.

[108] D. Klapholz and A. Lockman, "Contextual reference resolution," *American Journal of Computational Linguistics, microfiche 36*, 1975.

[109] D. Carter, "A shallow processing approach to anaphor resolution," *Ph.D. thesis, University of Cambridge*, 1986.

[110] ——, "Common sense inference in a focus-guided anaphor resolver," *Journal of Semantics, 4, 237-246*, 1987.

[111] B. Boguraev, "Automatic resolution of linguistic ambiguities," *TR-11, University of Cambridge Computer Laboratory, Cambridge*, 1979.

[112] Y. Wilks, "Preference semantics," *Stanford AI Laboratory memo AIM-206. Stanford University*, 1973.

[113] E. Rich and S. LuperFoy, "An architecture for anaphora resolution," *In the Proceedings of the Second Conference on Applied Natural Language Processing (ANLP 2), 18-24*, 1988.

[114] K. Wittenburg, "A parser for portable nl interfaces using graph-unification-based grammars," *In the Proceedings of AAA186*, 1986.

[115] L. D. Erman, P. E. London, and S. F. Fickas, "The design and an example use of hearsay-iii," *In the Proceedings of IJCAI 7*, 1975.

[116] J. G. Carbonell and R. D. Brown, "Anaphora resolution: a multi-strategy approach," *In the Proceedings of the 12th International Conference on Computational Linguistics (COLING 88), pp. 96-101*, 1988.

[117] B. Webber and R. Reiter, "Anaphora and locial form: On formal meaning," *In the Proceedings of the Fifth IJCAI, pp. 121-131*, 1977.

[118] J. G. Carbonell, "Discourse pragmatics in task-oriented natural language interfaces," *In the Proceedings of the 21st annual meeting of Association for Computational Linguistics (ACL 83)*, 1983.

[119] M. Tomita and J. G. Carbonell, "The universal parser architecture for knowledge-based machine translation," *In the Proceedings of the of IJCAI-87*, 1987.

[120] J. Bresnan and R. Kaplan, "Lexical-functional grammar: A formal system for grammatical representation," *The Mental Representation of Grammatical Relations. MIT Press, Cambridge, Massachusetts, pp. 173-281*, 1982.

[121] I. Dagan and A. Itai, "Automatic processing of large corpora for the resolution of anaphora references," *In the Proceedings of the 13th International Conference on Computational Linguistics (COLING 90) Vol. III, 1-3*, 1990.

[122] ——, "A statistical filter for resolving pronoun references," *In Feldman, Y.A. and Bruckstein, A. (Eds.) Artificial intelligence and computer vision, pp. 125-135*, 1991.

[123] K. Jensen, "Peg 1986: a broad-coverage computational syntax of english," *Technical report, IBM T.J. Watson Research Center*, 1986.

[124] S. Lappin and H. Leass, "An algorithm for pronominal anaphora resolution," *Computational Linguistics, 20(4), 535-561*, 1994.

[125] A. Voutilainen, J. Heikkil, and A. Anttila, "A constraint grammar of english: a performanceoriented approach," *Publication No. 21, Helsinki: University of Helsinki*, 1992.

[126] F. Karlsson, J. Heikkil, and A. Anttila, "Constraint grammar: a language-independent system for parsing free text," *Berlin/New York: Mouton de Gruyter*, 1995.

[127] B. Grosz, J. Aravind, and S. Weinstein, “Centering: a framework for modelling the local coherence of discourse,” *Computational Linguistics, 21 (2), 203-225*, 1995.

[128] M. A. Walker and A. K. Joshi, “Centering, anaphora resolution, and discourse structure,” *Oxford University Press, Oxford*, 1997.

[129] R. Mitkov, “Anaphora resolution in natural language processing and machine translation,” *Working paper. Saarbrücken: IAI*, 1995.

[130] ——, “An uncertainty reasoning approach for anaphora resolution,” *In the Proceedings of the Natural Language Processing Pacific Rim Symposium (NLPRS 5)*, 1995.

[131] ——, “Anaphora resolution: a combination of linguistic and statistical approaches,” *In the Proceedings of the 1st Discourse Anaphora and Anaphora Resolution Colloquium (DAARC 96)*, 1996.

[132] ——, “Introduction: Special issue on anaphora resolution in machine translation and multilingual nlp,” *Machine Translation, 14, 159-161*, 1999.

[133] ——, “Anaphora resolution,” *Computational Linguistics, 29(4)*, 2002.

[134] ——, “Discourse processing,” *The handbook of computational linguistics and natural language processing, pp. 599-629*, 2010.

[135] M. Poesio, R. Vieira, and S. Teufel, “Resolving bridging references in unrestricted text,” *In the Proceedings of a Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts (ANARESOLUTION 97), pp. 1-6*, 1997.

[136] M. Poesio, R. Mehta, A. Maroudas, and J. Hitzeman, “Corpus-based and computational approaches to discourse anaphora,” *In the Proceedings of the 42th Conference of the Association for Computational Linguistics (ACL 97)*, 1997.

[137] R. Vieira and M. Poesio, "A corpus-based investigation of definite description use," *Computational Linguistics, vol. 24, no. 2, pp. 183-216*, 1998.

[138] M. Poesio, T. Ishikawa, S. S. im Walde, , and R. Vieira, "Acquiring lexical knowledge for anaphora resolution," *In the Proceedings of the 3rd Conference on Language Resources and Evaluation (LREC 2002), pp. 1220-1224*, 2002.

[139] M. Marcus, B. Santorini, and M. Marcinkiewicz, "Building a large annotated corpus of english: the penn treebank," *Computational Linguistics, 19 (2), pp. 313-330*, 1993.

[140] M. Poesio, R. Mehta, A. Maroudas, , and J. Hitzeman, "Learning to resolve bridging references," *In the Proceedings of the 42th Conference of the Association for Computational Linguistics (ACL 2004)*, 2004.

[141] B. J. Grosz and C. L. Sidner, "Attention, intention, and the structure of discourse," *Computational Linguistics, 12(3):175-204*, 1986.

[142] J. R. Quinlan, "Programs for machine learning," *Morgan Kaufmann, San Francisco, CA.*, 1993.

[143] K. Church, "A stochastic parts program and noun phrase parser for unrestricted text," *In the Proceedings of the Second Conference on Applied Natural Language Processing, pp. 136-143*, 1988.

[144] D. M. Bikel and R. Schwartz, "An algorithm that learns what's in a name." *Machine Learning, 34(1-3) pp. 211-231*, 1999.

[145] G. A. Miller, "Wordnet: An on-line lexical database," *International Journal of Lexicography, 3(4) pp. 235-312*, 1990.

[146] V. Ng and C. Cardie, “Improving machine learning approaches to creference resolution,” *In the Proceedings of the 40th Conference of the Association for Computational Linguistics (ACL 2002), pp. 104-111*, 2002.

[147] D. L. Bean and E. Riloff, “Corpus-based identification of non-anaphoric noun phrases,” *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics, pp. 373-380*, 1999.

[148] ——, “Unsupervised learning of contextual role knowledge for coreference resolution,” *computational linguistics (HLT-NAACL 2004) pp. 297-304*, 2004.

[149] J. Allen, “Natural language understanding,” *Benjamin/Cummings Press, Redwood City, CA*, 1995.

[150] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the em algorithm.” *Journal of the Royal Statistical Society, 39(1), pp. 1-38*, 1977.

[151] C. Cherry and S. Bergsma, “An expectation maximization approach to pronoun resolution,” *In Proceedings of the 9th Conference on Computational Natural Language Learning (CoNLL 2005), pp 88-95*, 2005.

[152] N. Ge, J. Hale, and E. Charniak, “A statistical approach to anaphora resolution,” *In Proceedings of the Sixth Workshop on Very Large Corpora, pp. 161-171*, 1998.

[153] E. Charniak and M. Elsner, “Em works for pronoun anaphora resolution,” *In the Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2009) pp. 148-156*, 2009.

[154] E. Vorhees, “Overview of the trec 2002 question answering track.” *In Proceedings of the Eleventh Text Retrieval Conference (TREC 2002)*, 2002.

[155] S. Bergsma and D. Lin, “Bootstrapping path-based pronoun resolution,” *In the Proceedings of the 21st International Conference on Computational Linguistics (COLING 2006), pp. 33-40*, 2006.

[156] D. Lin, “Dependency-based evaluation of minipar.” *In the Proceedings of the Workshop on the Evaluation of Parsing Systems, First International Conference on Language Resources and Evaluation*, 1998.

[157] S. Bergsma, “Automatic acquisition of gender information for anaphora resolution.” *In the Proceedings of the Eighteenth Canadian Conference on Artificial Intelligence (Canadian AIX005), pp. 342-353*, 2005.

[158] H. Poon and P. Domingos, “Joint unsupervised coreference resolution with markov logic,” *In the Proceedings of the conference on Empirical Methods in Natural Language Processing (EMNLP 08) pp. 650-659*, 2008.

[159] A. Haghighi and D. Klein, “Unsupervised coreference resolution in a nonparametric bayesian model,” *In the Proceedings of the 45th Conference of the Association for Computational Linguistics (ACL 2007), pp. 848-855*, 2007.

[160] M. R. amd P. Domingos, “Markov logic networks,” *Machine Learning 62, pp. 107-136*, 2006.

[161] S. Kok, P. Singla, M. Richardson, P. Domingos, M. Sumner, H. Poon, and D. Lowd, “The alchemy system for statistical relational ai,” *http://alchemy.cs.washington.edu/*, 2007.

[162] N. Ge, J. Hale, and E. Charniak, “A statistical approach to anaphora resolution,” *In the Proceedings of the Sixth Workshop on Very Large Corpora, pp. 161-171*, 1998.

[163] D. McClosky, E. Charniak, and M. Johnson, “Bllip north american news text, complete,” *Linguistic Data Consortium. LDC2008T13*, 2008.

[164] A. Haghighi and D. Klein, “Coreference resolution in a modular, entity-centered model,” *In the Proceedings of Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2010.

[165] ——, “Simple coreference resolution with rich syntactic and semantic features,” *In the Proceedings of the Conference on EMNLP, pp. 1152-1161*, 2009.

[166] V. Stoyanov, N. Gilbert, C. Cardie, and E. Riloff, “Conundrums in noun phrase coreference resolution: Making sense of the state-of-the-art,” *In the Proceedings of the 47th Conference of the Association for Computational Linguistics (ACL 2009)*, 2009.

[167] A. Rahman and V. Ng, “Supervised models for coreference resolution,” *In the Proceedings of the conference on Empirical Methods in Natural Language Processing (EMNLP 09)*, 2009.

[168] K. Raghunathan, H. Lee, S. Rangarajan, N. Chambers, M. Surdeanu, D. Jurafsky, and C. Manning, “A multi-pass sieve for coreference resolution,” *In the Proceedings of the Conference on EMNLP, pp. 492-501*, 2010.

[169] MUC-6, “Coreference task definition,” *In the Proceedings of the Sixth Message Understanding Conference (MUC 6), pp. 335-344*, 1995.

[170] NIST, “The ace evaluation plan,” *In NIST*, 2004.

[171] J. Ghosh, “Scalable clustering methods for data mining,” *Handbook of Data Mining, chapter 10, pp. 247-277*, 2003.

[172] G. Durrett and D. Klein, “Easy victories and uphill battles in coreference resolution,” *In the Proceedings of the Conference on EMNLP*, 2013.

[173] S. Pradhan, L. Ramshaw, M. Marcus, M. Palmer, R. Weischedel, and N. Xue, "Conll-2011 shared task: Modeling unrestricted coreference in ontonotes," *In Proceedings of the Conference on Computational Natural Language Learning: Shared Task.*, 2011.

[174] E. Hovy, M. Marcus, M. Palmer, L. Ramshaw, and R. Weischedel, "Ontonotes: the 90 percent solution," *In the Proceedings of the Human Language Technology Conference of the NAACL, pp. 57-60*, 2006.

[175] G. R. Hofford, "Introduction to ross: A new representational scheme," *ArXiv e-prints, Nov. 2014*, 2014.

[176] A. Rahman and V. Ng, "Resolving complex cases of definite pronouns: The winograd schema challenge," *In the Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, (EMNLP-CoNLL 12), pp. 777-789*, 2012.

[177] I. R¨osiger and A. Riester, "Using prosodic annotations to improve coreference resolution of spoken text," *In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, pp. 83-88*, 2015.

[178] M. Strube and C. Müller, "A machine learning approach to pronoun resolution in spoken dialogue," *In Proceedings of the 41st Annual Meeting on Association for Computational Linguistics, pp. 168-175*, 2003.

[179] J. Tetreault and J. Allen, "Dialogue structure and pronoun resolution," *In the Proceedings of the 5th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC 2004)*, 2004.

[180] M. Amoia, K. Kunz, and E. L. Koltunski, "Coreference in spoken vs. written texts: a corpus-based analysis," *In the Proceedings of LREC, Istanbul*, 2012.

[181] J. Mayer, “Transcription of german intonation. the stuttgart system,” *University of Stuttgart*, 1995.

[182] A. Björkelund and J. Kuhn, “Learning structured perceptrons for coreference resolution with latent antecedents and non-local features,” *In the Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014), pp. 47-57*, 2014.

[183] K. Eckart, A. Riester, and K. Schweitzer, “A discourse information radio news database for linguistic analysis,” *In Sebastian Nordhoff Christian Chiarcos and Sebastian Hellmann, editors, Linked Data in Linguistics: Representing and Connecting Language Data and Language Metadata, pp. 65-76, Springer*, 2012.

[184] Y. Wilks, “Preference semantics,” *In Keenan, E. (Ed.) The formal semantics of natural language. Cambridge: Cambridge University Press*, 1975.

[185] R. Kantor, “The management and comprehension of discourse connection by pronouns in english,” *Ph.D. thesis, Department of Linguistics, Ohio University*, 1977.

[186] B. Grosz, “The representation and use of focus in a system for understanding dialogs,” *In the Proceedings of the 5th International Joint Conference on Artificial Intelligence (IJCAI 1977), pp. 67-76. Cambridge, Massachusetts*, 1977.

[187] F. Guenthner and H. Lehmann, “Rules for pronominalisation,” *In the Proceedings of the First Conference of the European Chapter of the Association for Computational Linguistics, pp. 144-151*, 1983.

[188] M. Rolbert, “Résolution de formes pronominales dans l’interface d’interrogation d’une base de données,” *Thèse de doctorat. Faculté des Sciences de Luminy*, 1989.

[189] A. Lockman, “Ph.d. thesis, contextual reference resolution,” *Faculty of Pure Science, Columbia University*, 1978.

[190] N. Asher and H. Wada, “A computational account of syntactic, semantic and discourse principles for anaphora resolution,” *semantic and discourse principles for anaphora resolution Journal of Semantics, 6, 309-344*, 1988.

[191] M. Kameyama, “Recognizing referential links: an information extraction perspective,” *In the Proceedings of the (ACL/EACL 97) Workshop on Operational Factors in Practical, Robust Anaphora Resolution, pp. 46-53*, 1997.

[192] S. Harabagiu and S. Maiorano, “Multilingual coreference resolution,” *In the Proceedings of ANLP/NAACL, pp. 142-149. Seattle, Washington*, 2000.

[193] C. Brew, D. McKelvie, R. Tobin, H. Thompson, and A. Mikheev, “The xml library lt xml,” *Version 1.2, LT-XML*, 2000.

[194] E. Charniak and M. Johnson, “Coarse-to-fine n-best parsing and maxent discriminative reranking,” *In the Proceedings of Association for Computational Linguistics*, 2005.

[195] C. Müller and M. Strube, “Multi-level annotation of linguistic data with mmax2,” *In S. Braun, K. Kohn, and J. Mukherjee, editors, Corpus Technology and Language Pedagogy: New Resources, New Tools, New Methods. Peter Lang, Vol. 3, pp. 197-214*, 2006.

[196] T. Kudoh and Y. Matsumoto, “Use of support vector machines for chunk identification,” *In the Proceedings of CoNLL*, 2000.

[197] J. R. Finkel, T. Grenager, and C. Manning, “Incorporating non-local information into information extraction systems by gibbs sampling,” *In the Proceedings of the 43rd Conference of the Association for Computational Linguistics (ACL 2005), pp. 363-370*, 2005.

[198] L. Qiu, M. Kan, and T. Chua, "A public reference implementation of the rap anaphora resolution algorithm," *In the Proceedings of the Fourth International Conference on Language Resources and Evaluation. Vol. I, pp. 291-294*, 2004.

[199] B. O'Connor and M. Heilman, "Arkref: a rule-based coreference resolution system," *arXiv:1310.1975, Oct 2013.*, 2013.

[200] V. Stoyanov, C. Cardie, N. Gilbert, E. Riloff, D. Buttler, and D. Hysom, "Coreference resolution with reconcile," *In the Proceedings of the 48th Conference of the Association for Computational Linguistics (ACL 2010)*, 2010.

[201] E. Bengtson and D. Roth, "Understanding the value of features for coreference resolution," *In the Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP 2008)*, 2008.

[202] L. Sobha and B. N. Patnaik, "Vasisth: An anaphora resolution system for indian languages," *In the Proceedings of International Conference on Artificial and Computational Intelligence for Decision, Control and Automation in Engineering and Industrial Applications*, 2000.

[203] ——, "Vasisth: An anaphora resolution system for malayalam and hindi," *In Proceedings of Symposium on Translation Support Systems*, 2002.

[204] R. Prasad and M. Strube, "Discourse salience and pronoun resolution in hindi," *Penn Working Papers in Linguistics, Vol 6.3, pp. 189-208*, 2000.

[205] B. Uppalapu and D. M. Sharma, "Pronoun resolution for hindi," *In the Proceedings of the 7th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC 09), pp. 123-134*, 2009.

[206] P. Dakwale, V. Mujadia, and D. M. Sharma, "A hybrid approach for anaphora resolution in hindi," *In the Proceedings of the International Joint Conference on Natural Language Processing, pp. 977-981*, 2013.

[207] K. Dutta, N. Prakash, and S. Kaushik, "Resolving pronominal anaphora in hindi using hobbs algorithm," *Web Journal of Formal Computation and Cognitive Linguistics, Issue 10, 2008*, 2008.

[208] K. N. Murthy, L. Sobha, and B. Muthukumari, "Pronominal resolution in tamil using machine learning approach," *The First Workshop on Anaphora Resolution (WAR I) , Ed Christer Johansson, Cambridge Scholars Publishing, 15 Angerton Gardens, Newcastle, NE5 2JA, UK, pp. 39-50*, 2008.

[209] L. Sobha and B. N. Patnaik, "Resolution of pronominals in tamil," *Computing Theory and Applicat ion, The IEEE Computer Society Press, Los Alamitos, CA, pp. 475-79*, 2007.

[210] K. N. Murthy, L. Sobha, and B. Muthukumari, "Conditional random fields based pronominal resolution in tamil," *International Journal on Computer Science and Engineering, Vol. 5 Issue 6 pp. 601-610*, 2013.

[211] J. Balaji, T. V. Geetha, R. Parthasarathi, and M. Karky, "Two-stage bootstrapping for anaphora res olution," *In the Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012), pp. 507-516*, 2012.

[212] P. Dakwale and H. Sharma, "Anaphora resolution in languages using hybride approaches," *In the Proceedings of the NLP Tool Contest, (ICON 2011)*, 2011.

[213] L. Sobha, V. S. Ram, and B. N. Patnaik, "A generic anaphora resolution engine for indian languages," *In the Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014), pp. 1824-1833*, 2014.

[214] U. K. Sikdar, A. Ekbal, and S. Saha, "Feature selection in anaphora resolution for bengali: A multiobjective approach," *In the Processing of Conference of Intelligent Text processing and Computational Linguistic, pp. 252-263*, 2015.

[215] S. Bhattacharya, M. Choudhury, S. Sarkar, and A. Basu, "Inflectional morphology synthesis for bengali noun, pronoun and verb systems," *In the Proceedings of the National Conference on Computer Processing of Bangla (NCCPB 2005), pp. 34-43*, 2005.

[216] S. K. Chatterji, "Bhasa prakash bangla byakaran (the grammar of the bangla language)," *Rupa publication, Calcutta*, 1993.

[217] A. Chakrabarty and U. Garain, "Benlem (a bengali lemmatizer) and its role in wsd," *In ACM Trans. Asian and Low-Resource Language Information Processing (TALIIP), Vol. 15, No. 3*, 2015.

[218] P. Majumder, M. Mitra, S. K. Parui, G. Kole, P. Mitra, and K. Datta, "Yass: Yet another suffix stripper," *ACM Transactions on Information Systems (TOIS), Volume 25 Issue 4, October 2007, Article No. 18*, 2007.

[219] S. K. Chatterji, "The origin and development of the bengali language," *Calcutta: Calcutta University Press*, 1926.

[220] S. Sen, "Bhasar itivritta (the history of language)," *Ananda publication, Calcutta*, 1993.

[221] P. Sarkar and G. Basu, "Bhasa jiggnasa (queries of language)," *Vidyasagar Pustak Mandir, Calcutta*, 1994.

[222] J. B. Chaki, "Bangla bhasar byakaran (the grammar of the bangla language)," *Ananda publication, Calcutta*, 1996.

[223] N. S. Dash, "Bangla pronouns: A corpus-based study," *Literary and Linguistic Computing. Vol. 15. No. 4. pp. 433-443*, 2000.

[224] S. Bhattacharya, M. Choudhury, S. Sarkar, and A. Basu, "Inflectional morphology synthesis for bengali noun, pronoun and verb systems," *In the Proceedings of the*

National Conference on Computer Processing of Bangla (NCCPB 2005), pp. 34-43, 2005.

[225] H. R. Thompson, "Bengali," *John Benjamins Publishing Company, Vol. 18.*, 2012.

[226] P. Rayson, D. Berridge, and B. Francis, "Extending the cochran rule for the comparison of word frequencies between corpora," *In the Proceedings of the 7th International Conference on Statistical analysis of textual data (JADT 2004), pp. 926-936*, 2004.

[227] M. Oakes, "Statistics for corpus linguistics," *Edinburgh: Edinburgh University Press, pp. 266*, 1998.

[228] P. Rayson and R. Garside, "Comparing corpora using frequency profiling," *In the Proceedings of the workshop on Comparing Corpora, in 38th Conference of the Association for Computational Linguistics (ACL 2000), pp. 1-6*, 2000.

[229] Z. Harris, "String analysis of the language structure," *Mutton and Co., The Hauge*, 1962.

[230] S. Klein and R. Simmons, "A computational approach to grammatical coding of english words," *Journal of the Association for Computing Machinery, 10: pp. 334-337*, 1963.

[231] B. B. Greene and G. M. Rubin, "Automatic grammatical tagging of english," *Technical Report, Department of Linguistics, Brown University*, 1971.

[232] D. Cutting, J. Kupiec, J. Pedersen, and P. Sibun, "A practical part-of-speech tagger," *In the Proceedings of the 3rd Conference on Applied Natural Language Processing, pp. 133-140*, 1992.

[233] B. Merialdo, "Tagging english text with a probabilistic model," *Computational Linguist, pp. 155-171*, 1994.

[234] ——, "Tagging english text with a probabilistic model," *Computational Linguist, pp. 155-171*, 1994.

[235] T. Brants, "Tnt a statistical parts-of-speech tagger," *In the Proceedings of the 6th Conference on Applied Natural Language Processing (ANLP 2000), pp. 224-231*, 1996.

[236] A. Ratnaparkhi, "A maximum entropy model for part-of-speech tagging," *In the Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 133-142*, 1996.

[237] J. D. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data." *In the Proceedings of the Conference ICML-2001, pp. 282-289*, 2001.

[238] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal 27 (3): pp. 379-423*, 1948.

[239] A. Berger, S. A. Pietra, and V. J. Pietra, "A maximum entropy approach to natural language processing," *Computational Linguistics, 22(1), pp. 39-71*, 1996.

[240] R. Lau, R. Rosenfeld, and S. Roukos, "Adaptive language modeling using the maximum entropy principle," *In the Proceedings of the Human Language Technology Workshop, pp. 108-113*, 1993.

[241] A. Bharati, D. M. Sharma, K. V. Ramakrishnamacharyulu, and R. Sangal, "Guidelines for anncorra: An introduction," *Technical Report TR-LTRC-14, Language Technologies Research Centre, IIIT Hyderabad*, 2001.

[242] B. B. Chaudhuri, N. S. Dash, and P. K. Kundu, "Computer parsing of bangla verbs," *In Linguistics Today, 1(1), pp. 64-86*, 1997.

[243] S. Dasgupta and M. Khan, “Feature unification for morphological parsing in bangla,” *In the Proceedings of the International Conference on Computer and Information Technology*, 2004.

[244] N. S. Das, “The morphodynamics of bengali compounds decomposing them for lexical processing,” *Language in India (www.languageinindia.com), pp. 6-7.*, 2006.

[245] S. Dasgupta and V. Ng, “Unsupervised morphological parsing of bengali,” *Language Resources and Evaluation, vol. 4, pp. 311-330*, 2006.

[246] K. Beesley and L. Karttuneni, “Finite state morphology,” *Computational Linguistics, CSLI Publications, Vol. 30, pp. 237-239*, 2003.

[247] I. Aduriz, E. Agirre, I. Aldezabal, I. Alegria, X. Arregi, J. M. Arriola, X. Artola, K. Gojenola, A. Maritxalar, K. Sarasola, and M. Urkia, “A word-grammar based morphologieal analyzer for agglutinative languages,” *In the Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000), pp. 1-7*, 2000.

[248] M. K. Anand, V. Dhanalakshmi, and U. R. Rekha, “A novel algorithm for tamil morphological generator,” *In the Proceedings of the ICON 2010, IIT Kharagpur*, 2010.

[249] C. D. Manning, P. Raghavan, and H. Schütze, “Chapter 8 evaluation in information retrieval,” *An Introduction to Information Retrieval Online edition (c) 2009 Cambridge UP*, 2009.

[250] B. B. Chaudhuri and S. Bhattacharya, “An experiment on automatic detection of named entities in bangla,” *In the Proceedings of the Workshop on NER for SASEAL in IJCNLP-08, pp. 85-91*, 2006.

[251] A. Ekbal and S. Bandyopadhyay, “A conditional random field approach for named entity recognition in bengali and hindi,” *Linguistic Issues in Language Technol-*

ogy (LiLT), Volume (2:1), November 2009, PP.1-44, CSLI Publication, Stanford University, 2009.

[252] ——, "Named entity recognition using appropriate unlabeled data, post-processing and voting," *In Informatica, Volume (34), Number (1), pp. 55-76*, 2010.

[253] ——, "A mul-tiengine ner system with context pattern learn-ing and post-processing improves system perfor-mance," *International Journal of Computer Processing of Languages (IJCPOL), World Scientific Press, Singapore, Volume (22:2-3), PP. 171-204*, 2010.

[254] A. Senapati and U. Garain, "A maximum entropy based honorificity identification for bengali pronominal anaphora resolution," *In the Proceedings of the 15th Conference of Intelligent Text processing and Computational Linguistics (CICLing 2014), pp. 319-329.*