ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 8(27), DOI:10.17485/ijst/2015/v8i27/83320, October 2015* ISSN (Online) : 0974-5645

Resolution of Anaphors in Punjabi

Singh*

Jawaharlal Nehru University, New Delhi – 110 067, India; harjitsingh.jnu@gmail.com

Abstract

This paper presents a study of Punjabi anaphors and their automatic resolution. This is probably the first work of its kind in Punjabi that resolves lexical anaphors using POS tagged corpus of the language. A system called PARShas been worked out through an algorithm. The algorithm works on the grammatical features of the language captured through the tagged labels.

Keywords: Anaphora, Lexical Anaphors, PARS, Resolution

1. Introduction

Anaphora resolution is a computational process, which identifies antecedent in a text for exposing the whole reference of the whole text and employed for other tasks (e.g. Machine translation). Conventionally, anaphora resolution plays a vital role in diverse Natural Language Processing (NLP) applications such as information extraction, automatic abstracting, dialogue systems, question answering etc.

In Punjabi, this is an initial effort on automatic anaphora resolution. The corpus used is the Indian Language Corpus Initiatives (ILCI) parallel corpus which is being developed in 17 Indian languages including English. Approximately 5,000 sentences (from health and tourism domains of Punjabi Corpora) have been manually tagged by MSRI tool. At the point of anaphora resolution, only 1,000 POS tagged sentences used for resolving the problem of lexical anaphora resolution.

2. Reflexives and Reciprocals in Punjabi

In general, reflexives and reciprocals both are one sorts of pronominal system of Punjabi. In the context of reflexives, there are two forms (bare reflexive and possessive reflexive) in Punjabi that aremorphological inflected for cases (give foot note) and also occur in local as well as long distance domain of a sentence. In addition, like other languages (Hindi, Malayalam, English etc.) by following some sources, Punjabi encodes reflexivity and reciprocity. On the other hand, polysemy appears among reflexives when they used as reduplicate.

2.1 Reflexives

In Punjabi, pure form of reflexive (*a: p* 'self') that is inflected for different cases and possessive reflexive (əpɳe) also formed by suffixation of (ɳe). Whereas composite reflexive (*ó* 'he' +*a:p* 'himself') is constructed by merging of (pronoun + bare). Correspondingly, compound verb construction establishes reflexivity also.

2.1.1 Bare Reflexive (a: p 'Self')

It is available in a data and does inflect for cases and syntactically it can occur in local as well as long-distance domain of a sentence. For example:

(1)
Ónekhʊɖ̪nũ
He-ERG self DAT
vekhɪɑ
see-3. PST-MSG
'He looked at himself'.

* *Author for correspondence*

In this instance (1) *'khʊɖ̤/ 'a: p'* occurs in object position and bind with its subject (Óne'he') in an intra-sentence.

2.1.2 Possessive Reflexive (ɘpɳa 'His / Her Own')

In Punjabi, 'ɘpɳɑ' is possessive reflexive that is inflected for direct / oblique cases and number and gender also. For example:

(2)
óəpɳeʈə̃
He-NOM his / her own PP
ɖərɖɑhɛ
scare-PRS-IPFV is-3. MSG
'He scares himself'.

In this instance (2) 'ɘpɳe' possessive reflexive inflected for number, gender and case and bind with it's antecedent (*ó* 'he').

2.1.3 Verbal Reflexive

In Punjabi, there is no available any specific affix that could represent reflexive construction however intransitive verbs are inherently take reflexive sense and able to form reflexive constructions. For example:

(3)
óɖo̯ɽɛvɑ
He-NOM ran-PST 3. MSG
'He run'. Lit. (He run himself).

In this instance (3) a verb *ɖo̯ɽɛvɑ'run'* inherently reflects reflexive sense on the agent third person (*ó* 'he').

2.2 Reciprocals

Punjabi has a core reciprocal form (*ɪkdʊje*, 'each other') that lexically formed by the combination of cardinal number (ɪk 'one') and ordinal number (*dʊje* 'two') but it does not inflect for phi features. By using different criteria of reciprocal construction (particular 'seven types of reciprocal marking' in Vladimir 2007: 150) I have tried to establish possible types of Punjabi reciprocals but here it does not matter of discussion. I am concentrating solely on the core reciprocal that syntactically occurs local as well as long distance domain of a sentence. For example:

(4)
riʈɑəʈ̪egitɑ ne
Rita and Geeta ERG
ik-ɖ̤uje nu vekʰɪɑ
each-other DAT see-PST 3. MSG
'Rita and Geeta saw each-other'

In this instance (4) reciprocal (*ɪkdʊje*'each-other') is in object position and bind with compound noun antecedent (*riʈɑəʈ̪egitɑ* 'Rita and Geeta') in a local domain of a sentence.

3. Punjabi Anaphora Resolution System (PARS)

Punjabi Anaphora Resolution System (PARS) is based on the java object-oriented programming language. This system runs on the PCs on the window by the utilization of java and jsp. This system is designed to take as input the annotated Punjabi text and resolve the lexical anaphors as specified in the algorithm. Algorithm, given below is a set of particular rules that solves the problem of a given domain. The algorithm for anaphora resolution is conceived for matching anantecedent and anaphor accurately and automatically with the help of the program. The system, however will resolve the lexical anaphors occurring intra-sentential.

3.1 Algorithm for Reflexive Anaphora

Tokenize the sentence (S) of the input text.
Find those (S) where PRF tags occur.
Find those (S) where NC, NP, PPR, PRL, DEM categories are preceded by PRF
See whether above NC, NP, PPR, PRL, DEM are followed below preference list.
Check all NP, NC, PPR, PRL or DEM has .nom or .dir tag.
Check all NP, NC, PPR, PRL or DEM has .nom or .obl tag.
Check all NP, NC, PPR, PRL or DEM which has .ins or .dir tag.
Check all NP, NC, PPR, PRL or DEM which has .ins or .obl tag.
Check all NP, NC, PPR, PRL or DEM has .dir 0 tag.
Check all NC, NP, PPR, PRL or DEM has .obl 0 tag.

The choice of antecedent will be based on the above preference list. If initially first preference list is fulfilled then any one of them (NC, NP, PPR, PRL, DEM) will be selected as antecedent for PRF.

To continue this process until above preference list does not fulfill. If two or more categories have same attributes then select nearest one as antecedent for PRF.

3.1.1 Algorithm for Reflexive Cataphora

If the S does not have any preceding NP, NC, PPR, PRL or DEM categories then try to find these categories in the following position of PRF.

If NP, NC, PPR, PRL or DEM any of them occur in the succeeding position of S and contain all attributes then consider it as antecedent.

Select nearest one if more than one category is followed PRF.

3.2 Algorithm for Reciprocal Anaphora

Tokenize the sentence (S) of the input text.
Find those (S) where PRC tags occur.
Find those (S) where NC, NP, PPR, PRL or DEM categories are preceded by PRC.
See whether above NC, NP, PPR, PRL, DEM are followed below preference list.
Check all NP. NC, PPR, PRL or DEM has .nom or .dir tag.
Check all NP, NC, PPR, PRL or DEM has .nom or .obl tag.
Check all NP, NC, PPR, PRL or DEM has .ins or .dir tag.
Check all NP, NC, PPR, PRL or DEM has .ins or .obl tag.
Check all NP, NC, PPR, PRL or DEM has .dir 0 tag.
Check all NP, NC, PPR, PRL or DEM has .obl 0 tag.

The choice of antecedent will be based on the above preference list. If initially first preference list is fulfilled then any of them (NP, NC, PPR, PRL, DEM) will be selected as antecedent for PRC.

To continue this process until above preference list does not fulfill.

If two or more categories have same attributes then select nearest one as antecedent for PRC.

3.2.1 Algorithm for Reciprocal Cataphora

If the S does not have any preceding NP, NC, PPR, PRL or DEM categories then try to find these categories in the following position of PRC.

If NP, NC, PPR, PRL or DEM any one of them occur in the succeeding position of S and contain all attributes then consider it as antecedent.

Select nearest one if more than one category is followed PRC.

3.3 The Usability of the PARS

Our system takes an input as POS tagged data and tokenizes with the help of delimiter a singled and a (dandi = vishramchinh) because in the Punjabi language, the completion of a sentence is marked by a dandi. Another string tokenization module extracts the words according to the presence of space between words. Third module presents the co-reference condition between anaphors and antecedents.

3.4 Limitations of the System

This system does not resolve all sorts of anaphors in Punjabi. It's primarily aim is to find the antecedent for an anaphor within a sentence. It does not resolve those anaphors that are occurred in inter-sentential. It does not resolve two PRF entities in the same sentence at the same time. It does not resolve accurately all PRF when they need DEM category as antecedent. All these restrictions are exceptional for our system because out of 1,000 sentences, there is solely two or three occurrence of the sentences where DEM category does not get antecedent for PRF but this is not an unresolved issue.

4. Further Developments

In future, this system can be more robust by some modifications in the algorithm. One of the enormous issues here are to resolve inter-sentential lexical anaphors. As per the algorithm, if more than one NC or NP etc. are within sentence and then nearest category will be the antecedent for PRF. This condition can be modified to include distanced and scrambled categories as well. However this will require many other critical changes like identifying the phrases, sentences and discourse elements correctly.

5. References

1. Carter D. Interpreting anaphora in natural language texts. Chichester: Ellis Horwood; 1987.
2. Carbonell J, Brown R. Anaphora resolution: A multi-strategy approach. Proceedings of the 12th International Conference on Computational Linguistics; 1988. p. 96–101.
3. Franz A. Automatic anaphora resolution in natural language processing: An empirical approach. Tokoyo: Springer; 2001.
4. Gopal M. Dissertation of computational anaphora and cataphora in the Sanskrit text Panchtantra, centre for linguistics. New Delhi: Jawaharlal Nehru University; 2011.
5. Hicks G. The derivation of anaphoric relations. Amsterdem: John Benjamins Publishing Company; 2009.

6. Huang Y. Anaphora: A cross-linguistic study. Oxford. New York: Oxford University Press (OUP). 2000.
7. Jha GN. The TDIL program and the Indian Language Corpora Initiative (ILCI). Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10); Malta: ELRA. 2010. p. 982–5.
8. Konig E, Gast V, editors. Reciprocals and reflexives: Theoretical and typological explorations. New York: Mouton de Grutyer; 2008.
9. Mitkov R. Anaphora resolution. London: Longman; 2002.
10. Poesio M, et al. Computational models of anaphora resolution: A survey. University of Heidelberg; 2010.
11. Wilcock G. Introduction to linguistic annotation and text analytics. Morgan and Claypool Publishers; 2009.
12. Zygmunt F, Traci C. Reflexives: Forms and Functions. Amsterdem: Benjamins; 1999.