

Anaphora Resolution in Multi-Person Dialogues

Prateek Jain and Manav Ratan Mital and Sumit Kumar and Amitabha Mukerjee and Achla M. Raina
Indian Institute of Technology Kanpur,
Kanpur 208016 INDIA
{pjain, sumit, manavrm, amit, achla}@iitk.ac.in

Abstract

Anaphora resolution for dialogues is a difficult problem because of the several kinds of complex anaphoric references generally present in dialogic discourses. It is nevertheless a critical first step in the processing of any such discourse. In this paper, we describe a system for anaphora resolution in multi-person dialogues. This system aims to bring together a wide array syntactic, semantic and world knowledge based techniques used for anaphora resolution. In this system, the performance of the heuristics is optimized for specific dialogues using genetic algorithms, which relieves the programmer of hand-crafting the weights of these heuristics. In our system, we propose a new technique based on the use of anaphora chains to enable resolution of a large variety of anaphors, including plural anaphora and cataphora.

1 Introduction

Anaphoric references abound in natural language discourses and their resolution has often been identified as the first step towards any serious discourse processing related tasks. However, any comprehensive anaphora resolution scheme is expected to entail the use of rich semantic and pragmatic knowledge representation and processing, and is, therefore, a complex problem. As a result of such problems, several heuristics-based approaches have been developed and adopted over the years to achieve partial solutions to the problem.

The pioneering work in the area of anaphora resolution was done by Hobbs (Jerry R. Hobbs, 1978) who designed several early syntactic and semantic heuristics for the same. (Hirst, 1981) discusses several early approaches to anaphora resolution in discourses. (Denber,

1998) and (Lappin and Leass, 1994) describe several syntactic heuristics for reflexive, reciprocal and pleonastic anaphora, among others. Often domain-specific heuristics are used for anaphora resolution and fine tuned to perform well on a limited corpus, such as in (Mitkov, 1998). (Ng and Cardie, 2002) proposes a machine learning approach to Anaphora Resolution but generally statistical learning approaches suffer from the problems of small corpuses and corpus dependent learning. A more general and comprehensive overview of state-of-the-art in anaphora resolution is given in (Mitkov, 1999) and also in (Mitkov et al., 2001).

Few systems have been developed that are specifically aimed at the task of anaphora resolution in discourses. ROSANA, an algorithm for anaphora resolution that focuses on robustness against information deficiency in the parsed output, is described in (Stuckardt, 2001). MARS, the Mitkov Anaphora Resolution System, is another automatic, knowledge-poor anaphora resolution system that has been implemented for several languages including English, Bulgarian and Japanese.

In this paper, we describe the design and implementation of *Jepthah*¹, a rule-based system for resolving a wide variety of anaphora occurring in multi-person dialogues in English. In this system, we integrate several different knowledge-poor constraints and heuristics, and operate them over a naive character model of the entire dialogue to perform effective anaphora resolution. In addition to using standard heuristics, we have developed our own semantic and pragmatic heuristics, specific to dialogue situations, that operate on this character model. There is a weight assigned to each of these heuristics and these weights are fine-tuned using a learning mechanism implemented by genetic algorithms. We use the linguistic feature of *anaphoric chains*, present in dialogues, to resolve a relatively large class of anaphora.

¹name of a wise Israeli judge in the Bible

2 Jephthah

In *Jephthah*, we adopt an integrated approach towards resolving various different kinds of anaphors occurring in dialogue situations. In this approach we fuse together several heuristics with a new kind of computational linguistic insight – that of the deployment of *anaphora chains* and we develop a graph-based technique for handling the resolution of various anaphors. An anaphora chain may be described as a referential chain comprising series of mutually co-referential anaphoric elements, generally of more than one type, headed by a referential element.

The class of anaphors that we aim to resolve is fairly large and includes pronouns, reflexives and deictic anaphors. In terms of distribution, we are dealing with anaphors in subject, object and modifier positions, possessive reflexive, and cataphora. It may be mentioned here that we deal only with unambiguous cases of plural pronouns, such as *both of us*, *two of you*, etc. These are the cases in which the domain of the pronouns is clearly quantified, unlike the case of such instances as *all of us* or *they*, etc.

2.1 Graph-theoretic Approach

The entire operation is centered around a graph formulation of the resolution problem in the perspective of the dialogue. We extract all the nouns and pronouns present in the dialogue. Assume there are n nouns and p pronouns in the dialogue. Let the i^{th} noun be represented as N_i , with $i \leq n$ and that P_i represents the i^{th} pronoun, with $i \leq p$. Now, we construct a graph representation for the problem as follows. Let G be the graph that we are interested in formulating, comprising of a node for every N_i and P_j . Let N_i^G be the node corresponding to the noun N_i and P_j^G be the node corresponding to the pronoun P_j . Thus, we can split the set of vertices of this graph V_G into two parts, the set consisting of $N_i^G, \forall i \leq n$ and the set consisting of $P_j^G, \forall j \leq p$. The set of edges E_G for this graph G comprises of two types of **directed edges** and is constructed as follows. Construct a set of edges \mathcal{E}_1 which includes a directed edge $E_{i \rightarrow j}$ from P_i^G to N_j^G , for all pairs P_i^G and N_j^G . The other set \mathcal{E}_2 includes a directed edge $E'_{i \rightarrow j}$ from P_i^G to P_j^G for all pair of nodes P_i^G and P_j^G such that $i \neq j$. Clearly, we have $E^G = \mathcal{E}_1 \cup \mathcal{E}_2$. Let us define a property \mathcal{L} on the paths in this graph as follows – a path p satisfies the property \mathcal{L} iff it consists of a sequence of edges $E_i \in E_G$ ($i \leq \text{length}(p)$) with exactly one edge E_f from the set \mathcal{E}_1 and the condition that this is the last edge in the sequence, i.e., $E_{\text{length}(p)} \equiv E_f$.

Intuitively, this graph represents the set of possible anaphor-antecedent relationships. The set of possible referents of an anaphor represented by the node P_i^G in the graph G consists of all possible distinct nodes N_k^G that

can be reached from P_i^G using paths that satisfy the property \mathcal{L} . Let this set be represented as S_i . Note here that paths as above of length ≥ 2 represent anaphoric chains present in the dialogue. One or more edges in these paths are from one anaphor to another and represent co-reference amongst these anaphors. The *antecedent space* of an anaphor P_i consists of all nouns and pronouns whose corresponding nodes in the graph G are reachable from P_i^G by traversing a single edge belonging to E_G . Now, the idea here is to process this antecedent space and rank all the nodes in S_i to determine the most likely antecedent for the anaphor P_i . This ranking is done by attaching weights to the edges present in the graph.

Every edge is awarded a particular weight (less than 1.0), that is evaluated for every edge using a set of heuristics described in section 2.4. The rank of each node N_k^G in the set S_i is determined by the total weight W_{ik} for that node. W_{ik} is computed as follows – let the weight W_p of each path p be defined as the product of the weights of all the edges lying on that path. Then, W_{ik} is the sum of the weights of all the paths from P_i^G to N_k^G , i.e., $\sum_p W_p$. Hence, for anaphora resolution, we need to basically design an algorithm or a function to compute the weight for each edge in the graph.

2.2 System Design

The input dialogue is passed to the front end which comprises of the Stanford Serialized Parser and PoS tagger. The parser gives the best parse for every input sentence, each of which are then subsequently processed. In the first step we extract all the proper nouns present in the dialogue and initialize our character model base and the graph G that was explained in section 2.1. We then take the sequence of parses corresponding to each subsequent dialogue by a speaker and process them sequentially. Techniques for anaphora resolution are then applied in two phases. In the first phase, a set of *constraints* is applied to this graph, to prune out edges that represent any unfeasible co-references. In the second phase, a set of *heuristics* are applied to award weights to edges representing these relationships. After the processing is over and all weights have been obtained, the permissible antecedents for each anaphor are ranked and the most likely antecedent for each is outputted. In case there is a plural anaphor, with quantification over x nouns, the top x likely antecedents are outputted.

While processing the dialogue, a naive character building is implemented. This is done mainly by focusing on the verbs in the sentences. The subject and object nouns associated with these verbs are selected and their relationship is put in the character model base associated with the speaker of the corresponding dialogue. The system maintains an apriori knowledge base with it containing information like ontology and functionalities of several nouns.

This combination of apriori and assimilated knowledge is then used to apply certain semantic and pragmatic constraints/heuristics on the graph, as shown in the following sections.

2.3 Constraints

We apply the set of restrictions prior to the set of preferences, thereby narrowing down the candidate set as early as possible. The list of constraints that implement these restrictions in *Jepthah* are listed as follows –

1. **Deictic Constraint:** This is a set of simple constraints that are specific to dialogue settings because in such settings we can have the concept of frames of reference with regard to the various speakers involved in the dialogue action.
2. **Non-coreference (Mitkov, 1999):** Syntactic features present in a sentence often lend themselves to be expressed as constraints on anaphora reference. These features are captured by our non-coreference constraints which stipulate that certain pairs of anaphor and noun phrases within the same sentence cannot refer to the same antecedent.
3. **Gender, Number and Person Agreement:** This is a low level constraint which requires that anaphors and their antecedents must agree in gender, number and person respectively.
4. **Constraint on Reflexive Pronoun:** A reflexive pronoun such as *himself*, *herself*, etc must refer to the subject or the object of the verb in whose clause it lies. In case of ellipsis, however, it may refer to the subject or object of the next higher verb to which the clause is attached.
5. **Semantic Consistency (Mitkov, 1999):** This constraint enforces same semantics of the antecedent as the anaphor under consideration.

2.4 Heuristics

Each preference or heuristic, has a certain weight and awards certain points to every anaphor-antecedent relationship. These points are a measure of the likelihood of that anaphor-antecedent relationship. The weight of an edge is the sum total of the weights awarded by each individual heuristic to the anaphor-antecedent relationship. The heuristics used in our system are enumerated as follows –

1. **Definiteness (Lappin and Leass, 1994):** According to this heuristic, nouns that are preceded by a demonstrative pronoun or a definite article are more likely to be antecedents and are awarded higher credibilities.

2. **Non-prepositional NP (Lappin and Leass, 1994):**

This heuristic states that a noun phrase which occurs within a prepositional phrase is less probable to be an antecedent to an anaphor and consequently, it is awarded less credibility.

3. **Pleonastic (Lappin and Leass, 1994):** This heuristic is based on the observation that there exist some syntactic patterns such that every *it* anaphor occurring in any of those patterns must be pleonastic.

4. **Syntactic Parallelism (Lappin and Leass, 1994):** As per this heuristic, preference is given to noun phrases with the same syntactic function as the anaphor.

5. **Recency (Mitkov, 1999):** This is a very simple heuristic according to which, everything else being comparable, a higher credibility is awarded to the antecedent nearer to the anaphor.

6. **Semantic Parallelism (Lappin and Leass, 1994):** This heuristic gives preference to those noun phrases which have the same semantic role as the anaphor in question. This is a useful heuristic and can be implemented by a system that can identify semantic roles.

7. **Pragmatic Heuristics:** We use certain pragmatic heuristics that we have identified to be very specific to dialogue settings. These are of the following kinds

- If one speaker asks a question, then the next speaker is likely to be the antecedent of the *you* that may occur in the former's sentence.
- If a speaker makes an exclamation then he is likely to be the antecedent of the *you* in the speech of the speaker just before him.

8. **Naive Character Building:** This refers to a naive character model that we have used to implement a restricted knowledge-based representation of the dialogue, woven around all the noun entities that are present in the dialogue. To this end, we use a certain amount of world knowledge that is present apriori with the system, in the form of ontology and functionality of possible noun entities. For instance, we associate actions with each character based on their subject object relationship with the verbs that occur in the dialogues. Now for an anaphor we see if a possible antecedent has functionality of the action associated with the anaphor, implied by the verb of the sentence. if it is so, we then give higher credibility to this particular antecedent.

Table 1: Results

Corpus	% Accuracy
Shaw's play - Pygmalion	62
Shaw's play - Man and Superman	67
Hand-Crafted Dialogue I	83
Hand-Crafted Dialogue II	81

2.5 Learning approach

In most systems ((Mitkov, 1998),(Lappin and Leass, 1994)) the weights that are assigned for different anaphor-antecedent relationships are programmer dependent. Fixing these values in an adhoc fashion can clearly give rise to unstable behaviour. In our work, we use manually tagged corpora to evaluate the effectiveness of a given weight assignment; these can then be tuned using Genetic Algorithms(Goldberg, 1989). We use 2-point crossover and mutation which are used in Standard Genetic Algorithm for Real Variables(Deb and Kumar, 1995).

3 Results

We used our system for anaphora resolution in the following types of dialogue corpora:

- Dialogues written manually, woven broadly in a student environment
- Fragments from the plays by the writer G. B. Shaw

Our System gave nearly 65% accuracy on Shaw's plays and almost 80% accuracy on our own "hand crafted" dialogues [Table:1]. In the table, the name "hand-crafted dialogues" refers to sample dialogues that the authors wrote themselves to test the performance of the system.

The genetic algorithms that we use help in fine-tuning weights according to the particular corpus, and show appreciable increase in accuracy.

4 Conclusions

We have implemented an automatic, knowledge-based anaphora resolution system that works for dialogic discourses. The lack of availability of any standard corpora (Mitkov, 1999) is a major drawback in case of anaphora resolution systems in general and those for dialogues in particular. The original contribution of this system is mainly two-fold. First, the anaphora resolution system that we have implemented uses an innovative graph technique, based on the idea of anaphora chaining, that makes it possible to resolve such references as cataphora and plural anaphora. Secondly, we give an algorithm which uses naive character building to apply various semantic and world-knowledge based heuristics to the process of

anaphora resolution. The results obtained from the system indicate a fairly high accuracy, though an extensive evaluation of the various resolution algorithms as well as the system as a whole remains to be done.

References

- K. Deb and A. Kumar. 1995. Real-coded genetic algorithms with simulated binary crossover: Studies on multimodal and multiobjective problems. *Complex Systems*, 9(6):431–454.
- M. Denber. 1998. Automatic resolution of anaphora in english. Technical report, Eastman Kodak Co., Imaging Science Division.
- D. E. Goldberg. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Publishing Company, Reading, MA.
- Graeme Hirst. 1981. Discourse-oriented anaphora resolution in natural language understanding: A review". *American Journal of Computational Linguistics*, 7(2):85–98, April-June.
- Jerry R. Hobbs. 1978. Resolving pronoun references. *Lingua*, 44:311–338.
- Shalom Lappin and Herbert J. Leass. 1994. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–561.
- Ruslan Mitkov, Branimir Boguraev, and Shalom Lappin. 2001. An Introduction to the Special Issue on Computational Anaphora Resolution. *Computational Linguistics*, 27(4).
- Ruslan Mitkov. 1998. Robust pronoun resolution with limited knowledge. In *COLING-ACL*, pages 869–875.
- R. Mitkov. 1999. Anaphora Resolution: The State of the Art. Working paper (Based on the COLING'98/ACL'98 tutorial on anaphora resolution).
- Vincent Ng and Claire Cardie. 2002. Combining sample selection and error-driven pruning for machine learning of coreference rules. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics*.
- Roland Stuckardt. 2001. Design and Enhanced Evaluation of a Robust Anaphora Resolution Algorithm. *Computational Linguistics*, 27(4):479–506, December.