

Antelogue: Pronoun Resolution for Text and Dialogue

Eleni Miltsakaki

University of Pennsylvania

elenimi@seas.upenn.edu

Abstract

Antelogue is a pronoun resolution prototype designed to be released as off-the-shelf software to be used autonomously or integrated with larger anaphora resolution or other NLP systems. It has modules to handle pronouns in both text and dialogue. In *Antelogue*, the problem of pronoun resolution is addressed as a two-step process: a) acquiring information about properties of words and the entities they represent and b) determining an algorithm that utilizes these features to make resolution decisions. A hybrid approach is implemented that combines known statistical and machine learning techniques for feature acquisition and a symbolic algorithm for resolution.

1 Introduction

Pronoun resolution is the well-known problem of identifying antecedents for pronominal references in text or dialogue. We present a prototype of new system for pronoun resolution, *Antelogue*, that handles both text and dialogues. In our approach, pronoun resolution is done in two steps: a) feature acquisition of properties of words and the entities they represent and b) resolution algorithm. We adopt a hybrid approach to the problem, using statistical and machine learning techniques widely available in the NLP literature to collect features and a symbolic algorithm informed by prior research in anaphora resolution and models of entity salience to appropriately rank and evaluate antecedents.

The design and architecture of *Antelogue* is modular and flexible and will soon be released for off-the-shelf use as an independent component or for possible integration of larger anaphora resolution systems, such as the GuiTAR (General Tool for Anaphora Resolution) (Poesio and Kabadjov, 2004) that currently is released with

(Mitkov et al., 2002)’s statistical pronoun resolution algorithm, MARS, that processes pronouns in text. Motivation for building a new algorithm for text and dialogues has been the problem of alignment between caption dialogues and stage directions on one hand and video content in movies on the other. While pronoun resolution in stage directions proved to be a fairly easy task, in dialogues we are facing the following challenges:

1. Part of speech taggers trained on text (typically the Wall Street Journal texts of Penn Treebank) perform poorly on dialogues, primarily due to the fragmented nature of spoken language. As a result NP tags are overgenerated.
2. Fragmentary speech and disfluencies or false starts common in dialogues cannot be handled by parsers trained on text.
3. First and second person pronouns are common. Special algorithms are needed to handle them.
4. Special *addressee* patterns need to be identified to block first and second person named references (e.g., “Hey, John, where did he go?”) becoming antecedents for third person pronouns.
5. In dialogues, pronouns can be used for reference to people or objects that are visually but not textually accessible. Special algorithms are needed to identify when an antecedent is not present in the text.
6. Pronouns are used for reference to people or objects that are visually salient in the scene but not mentioned explicitly in the dialogue, i.e., there are no textual antecedents.
7. Multi-party dialogues, sometimes 3rd person pronouns are used to refer to other speakers. It is hard to identify when an instance of a 3rd person pronoun has an antecedent in the prior discourse

or another speaker.

In what follows, we present the system’s design and architecture and the components that have already been implemented. In the demo, the users will be able to use *Antelogue*’s GUI to enter their own data and evaluate the system’s performance in real time. The current version handles first, second, and third person singular pronouns, including a classification recognizing referential and non-referential instances of “it”. *Antelogue* does not, yet, handle plural pronouns or recognize impersonal uses of singular “you”.

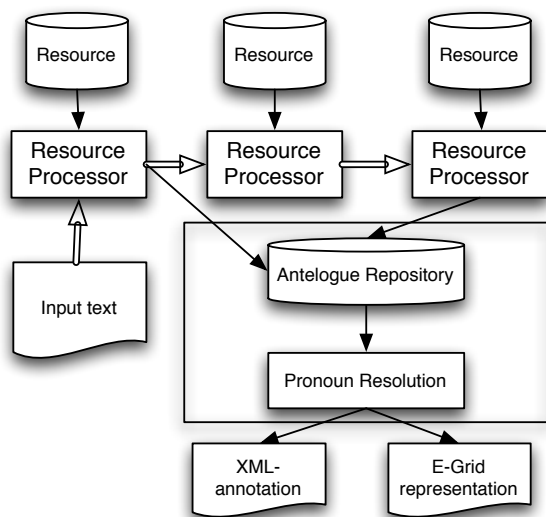


Figure 1: General System Architecture

2 System design

The problem of pronoun resolution is addressed as a two-step process: a) acquiring information about properties of words and the entities they represent and b) determining an algorithm that utilizes these features to make resolution decisions. A hybrid approach is implemented that combines known statistical and machine learning techniques for feature acquisition and a symbolic algorithm for resolution.

For the feature acquisition step, any number of feature acquisition sub-modules can be implemented. The architecture is flexible such that new feature acquisition modules can be added as they may become available or deemed crucial for specific applications. The demo version acquires fea-

tures from a sentence tokenizer, word tokenizer, NER tagger, gender and number database and POS tagger. For every sub-module a corresponding parser analyzes the output of the submodules to retrieve the features and store them in the *Antelogue repository*.

The resolution step implements an algorithm for utilizing the features in the repository to make resolution decisions. The resolution module needs to communicate only with the repository to get feature information and outputs xml annotated text or, what we call, *e-grid* output in which pronouns have been replaced by their antecedents. If the identified antecedent is a pronoun, it is further looked-up until a non-pronominal antecedent is found. A pronominal antecedent is shown only in case there is no nominal antecedent available.

The architecture of *Antelogue* is illustrated in Fig. 1. *Antelogue* can be set to perform pronoun resolution in both dialogue and text. A pre-processing step is required to ensure that the files are in the appropriate format. Because *Antelogue* was built to perform pronoun resolution in the dialogues and stage directions of screenplays, the pre-processing steps required to extract dialogues and text from the TV series *Lost*, are available.

3 System architecture

Feature acquisition *Sentence and word tokenization*: built based on (Ratnaparkhi, 1996). To address dialogue idiosyncrasies, sentence tokenization is forced to respect speaker turns thus blocking forming sentences across speaker turns.

Word processor. This module processes the word tokenized file and creates an indexed entry for every word in the *Antelogue repository*.

Named Entity Recognizer tagging (NER): We integrated Stanford’s NER tagger (Finkel et al., 2005).

NER processor. This module processor the NER tagged file and associates identified NER tags with the corresponding words in the *Antelogue repository*.

Gender and Animacy processor: This modules collects gender information from the gender corpus¹ (Bergsma and Lin, 2006) and checks a self-

¹<http://www.cs.ualberta.ca/~bergmsa/Gender>.

made corpus for profession (teacher, doctor, etc) and family relations (mother, father, etc), extracted from web searches. In the gender corpus, gender and number data are collected statistically and are not always reliable. We developed a simple confidence metric to evaluate the reliability of the gender and number data. If the ratio of the highest probability to the sum of all other probabilities is lower than 60% we mark gender or number unknown.² *Part-of-speech tagging (POS)*. We trained (Ratnaparkhi, 1996)’s POS tagger on dialogue data obtained from the English CTS Treebank with Structural Metadata released by LDC in 2009. *POS parser*. This module parses the POS-tagged input and updates the *Antelogue repository*.

Pronoun resolution The pronoun resolution submodule, currently, has three submodules: a) first and second person pronouns, b) third person singular masculine and feminine pronouns, and c) third person singular neuter pronouns.

For the first and second person pronouns, *Antelogue* identifies and resolves all instances of “I” to the speaker name and all instances of “you” to the next speaker. If there is no other speaker (when “you” is in the last turn), the algorithm will pick the speaker from the previous turn. If there is no previous turn, it is declared unresolvable.

For the third person “he” and “she” module, the algorithm *Antelogue* searches for pronouns backwards starting at the last sentence of the dialogue. For every sentence we construct a list of potential antecedents identified as nouns or pronouns by the POS tagger. A number of filters, then apply, to filter out incompatible antecedents. A category of incompatible antecedents for ‘he’ and ‘she’ that is almost unique to dialogues are addressee references. We identify references to addressee using surface punctuation features. Resolution starts with a look-up at antecedents of the current sentences, processing them from left-to-right. If the first antecedent is identified in the human corpus and has compatible gender information, it is picked. If not, the gender corpus is searched for reliable matches. Once a match is identified, it

is filtered by NER. The gender corpus often assigns feminine or masculine gender to common nouns. Only those entities that have a NER tag pass the compatibility test. If no compatible antecedent is found in the current sentence, *Antelogue* continues search in the previous sentence. If the dialogues have scene boundaries, as the case is in *Lost*, the search for an antecedents stops at a scene boundary. Otherwise it will not stop before the first sentence of the dialogue is reached. If no compatible antecedent is found, it is declared ‘unresolvable’. Correctly declaring pronouns unresolvable is extremely useful in dialogues, especially from movies, in which a referent of a third person pronoun may be visually available but not introduced in the prior discourse. Correctly unresolvable feminine and masculine pronouns signal a cue for search in the visual scene, a cross-modal direction that we are pursuing as part of future work.

For the third person “it”, we first need to address the issue of identifying referential and non-referential instances of “it”.³ Non-referential instances of “it” include pleonastic “it” (e.g., “it rains”, or “it is certain that...”) and references to a verbal or other clausal antecedent (e.g., “it” in “Mary got the award. It’s wonderful!”). For the “it” classification task, we follow (Bergsma et al., 2008)’ approach. We generate 4 and 5 word patterns out using the found occurrences of “it” then replace “it/its” with “they/theirs/them”. Frequencies of the substituted versions are computed using data from the Google n-gram corpus. If substitutions with “they/theirs/them” are not common, “it” is classified as non-referential.

Antelogue outputs a) an XML file with annotations of entities, pronouns and antecedents, and b) an “e-grid representation file” in which all pronouns have been replaced with their referents. In the XML file, pronouns are either resolved or declared unresolvable if no antecedent is identified. The pronoun “it” can, additionally, be declared non-referential. The e-grid representation file is useful for evaluating text coherence using the file directly as input to the (Barzilay and Lapata, 2008)’s e-grid model, a direction we want

²(Charniak and Elsnar, 2009)’s system ‘learns’ gender information using Expectation Maximization.

³For simplicity, we are sloppy here using the term non-referential to mean non-referring to a nominal entity.

to take in the future to explore its strengths in automatically identifying scene boundaries. Despite well-known problems in making meaningful comparisons in pronoun resolution systems, *Antelogue*'s performance is comparable to some of the highest reported performances, either identifying correctly an antecedent or correctly declaring a pronoun unresolvable or non-referential in 85% of 600 annotated pronouns.

Text module: *Antelogue*'s architecture for resolving pronouns in text is identical to dialogues except that a) the pre-processing text extracts text from the stage directions in the screenplay, b) addressee patterns are not used to filter out antecedents for "he" and "she" and instances of "I" and "you" are ignored. In the future we plan to implement resolution of "I" and "you" as well as a dialogue style resolution of "he" and "she" for instances of embedded speech. These instances were extremely rare in our data but they need to be catered for in the future. *Antelogue*'s performance exceeds 90% for stage directions because stage directions are relatively simple and fairly unambiguous. For this reason, a syntactic parse which slows down the system considerably was not used. However, to retain similar levels of performance in different domains, the use of syntactic parse will be needed.

4 Antelogue API and demo

Antelogue is implemented in Java. Its API includes an executable file, an empty database for the repository and command line instructions for running the system. The dialogue POS tagger is also available. The other feature acquisition sub-modules, text POS tagger, NER tagger and gender database are publicly available. *Antelogue* makes use of the google n-gram corpus, available through the Linguistic Data Consortium.⁴

As an off-the-shelf application, designed both for integration but also for experimentation, evaluation and comparison with other systems, *Antelogue* runs on a single unix command. The user is prompted to choose the dialogue or text module and then is asked to determine the path with the

data. *Antelogue* returns annotated files with resolved pronouns in seconds for a reasonably sized file (approx. 2,000-3,000 words) or in couple of minutes for very large files. These processing time estimates apply to the demo version. Processing time will vary depending on the number of submodule implemented in the feature acquisition step.

For the demo, we built a special Graphical User Interface. In the left part of the GUI, the user can either type in his or her own text or dialogue, paste text or dialogue, or select a local file. There are selections for the text/dialogue mode and xml/e-grid outputs. *Antelogue* performs pronoun resolution in real time and show the results on the right hand side part of the GUI.

Acknowledgments: Special thanks to Ben Taskar for his help and guidance in this project and to NSF IIS-0803538 grant for financial support.

References

- Barzilay, R. and M. Lapata. 2008. Modeling local coherence: An entity-based approach. *Computational Linguistics*.
- Bergsma, S. and D. Lin. 2006. Bootstrapping path-based pronoun resolution. In *ACL'06*, pages 33–40.
- Bergsma, S., D. Lin, and R. Goebel. 2008. Distributional identification of non-referential pronouns. In *ACL'08*, pages 10–18.
- Charniak, E. and M. Elsnar. 2009. Em works for pronoun resolution. In *Proceedings of EACL 2009*.
- Finkel, J.R., T. Grenager, and C. Manning. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. *Ann Arbor*, 100.
- Mitkov, R., R. Evans, and C. Orasan. 2002. A new, fully automatic version of Mitkov's knowledge-poor pronoun resolution method. *Lecture notes in computer science*, pages 168–186.
- Poesio, M. and M.A. Kabadjov. 2004. A general-purpose, off-the-shelf anaphora resolution module: Implementation and preliminary evaluation. In *Proc. of the 4th International Conference on Language Resources and Evaluation. Lisbon, Portugal*. Citeseer.
- Ratnaparkhi, A. 1996. A maximum entropy model for part-of-speech tagging. In *In Proceedings of EMNLP'96*, pages 133–142.

⁴<http://www ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2006T13>