Coreference Resolution for Italian: Assessing the Impact of Linguistic Components

Olga Uryupina
DISI - University of Trento
uryupina@gmail.com

Alessandro Moschitti
DISI - University of Trento
Qatar Computing Research Institute
moschitti@disi.unitn.it

Abstract

English. This paper presents a systematic evaluation of linguistic components required to build a coreference resolution system: mention detection and mention description. We compare gold standard annotations against the output of the modules based on the state-of-the-art NLP for Italian. Our experiments suggest the most promising direction for future work on coreference in Italian: we show that, while automatic mention description affects the performance only mildly, the mention detection module plays a crucial role for the end-to-end coreference performance. We also show that, while a considerable number of mentions in Italian are zero pronouns, their omission doesn't affect a general-purpose coreference resolver, suggesting that more specialized algorithms are needed for this subtask.

Italiano. *Questo articolo presenta una valutazione sistematica delle componenti linguistiche necessarie per costruire un sistema di risoluzione delle coreference: selezione automatica delle menzioni ad entitá e la loro descrizione. A questo scopo si confrontano le annotazioni gold standard contro l'output dei moduli basati sul NLP, che sono lo stato dell'arte per l'italiano. Questi esperimenti suggeriscono le direzione di ricerca piú promettenti per i futuri lavori su coreferenza in italiano: infatti, si dimostra che, mentre la descrizione automatica delle menzioni influisce sulle prestazioni solo leggermente, il modulo di selezione delle menzioni svolge un ruolo fondamentale per la prestazione del risolutore di coreference (end-to-end). Si dimostra anche che, mentre un numero considerevole di menzioni in italiano sono zero-pronouns, la loro omissione non pregiudica il risultato di coreferenza. Questo suggerisce che algoritmi piú specializzati sono necessari per questa sottoattivitá.*

1 Introduction

Coreference Resolution is an important prerequisite for a variety of Natural Language Processing tasks, in particular, for Information Extraction and Question Answering, Machine Translation or Single-document Summarization. It is, however, a challenging task, involving complex inference over heterogeneous linguistic cues. Several high-performance coreference resolvers have been proposed recently in the context of the CoNLL-2011 and CoNLL-2012 shared tasks (Pradhan et al., 2011; Pradhan et al., 2012). These systems, however, are engineered to process English documents and cannot be dirtily applied to other languages: while the CoNLL-2012 shared task includes Arabic and Chinese datasets, most participates have not investigated any language-specific approaches and have relied on the same universal algorithm, retraining it for particular corpora.

To our knowledge, only very few systems have been proposed so far to provide end-to-end coreference resolution in Italian. In the context of the SemEval-2010 shared task (Recasens et al., 2010), four systems have attempted Italian coreference. Among these toolkits, only BART relied on any language-specific solutions at this stage. The TANL system, however, was enhanced with language-specific information and integrated into the University of Pisa Italian pipeline later on (Attardi et al., 2012). At Evalita 2009 and 2011, different variants of coreference resolution were proposed as shared tasks (Lenzi and Sprugnoli, 2009; Uryupina and Poesio, 2012), in both cases, only one participant managed to submit the final run.

One of the bottlenecks in creating high-performance coreference resolvers lies in the complexity of their architecture. Coreference is a deep linguistic phenomenon and state-of-the-art

systems incorporate multiple modules for various related subtasks. Even creating a baseline end-to-end resolver is therefore a difficult engineering task. Going beyond the baseline is even more challenging, since it is generally unclear how different types of errors might affect the overall performance level. This paper focuses on systematic evaluation of different sub-modules of a coreference resolver to provide a better understanding of their impact on the system's performance and thus suggest more promising venues for future research. Starting with a gold pipeline, we gradually replace its components with automatic modules, assessing the impact. The ultimate goal of our study is to boost the performance level for Italian. We are focusing on improving the language-specific representation, leaving aside any comparison between coreference models (for example, mention-pair vs. mention-entity vs. graph-based).

2 Coreference Resolution Pipelines

End-to-end coreference resolvers operate on raw texts, requiring a full linguistic pipeline to preprocess the data. Below we describe the preprocessing pipeline used in our study and then proceed to the proper coreference pipeline.

2.1 Preprocessing pipeline

Our preprocessing pipeline for Italian is a part of the LiMoSINe project *Semantic Model Extractor*. The LiMoSINe Semantic Model contains various levels of linguistic description, representing a document from different angles. It should therefore combine outputs of numerous linguistic preprocessors to provide a uniform and deep representation of document's semantics. This raises the issue of the compatibility between such preprocessors: with many natural language processing (NLP) modules around, both publicly available and implemented by the LiMoSINe partners, it becomes virtually impossible to ensure that any two modules have the same input/output format and thus can be run as a pipeline. We have focused on creating an architecture that allows for straightforward incorporation of various tools, coordinating their inputs and outputs in a uniform way. Our Semantic Model is based on Apache UIMA— a framework for Unstructured Information Management, successfully used for a number of NLP projects, e.g., for the IBM Watson system.

TextPro wrapper. To provide basic levels of linguistic processing, we rely on TextPro—a suite of Natural Language Processing tools for analysis of Italian (and English) texts (Pianta et al., 2008). The suite has been designed to integrate various NLP components developed by researchers at Fondazione Bruno Kessler (FBK). The TextPro suite has shown exceptional performance for several NLP tasks at multiple EvalIta competitions. Moreover, the toolkit is being constantly updated and developed further by FBK. We can therefore be sure that TextPro provides state-of-the-art processing for Italian. TextPro combines rule-based and statistical methods. In addition, it allows for a straightforward integration of task-specific user-defined pre- and post-processing techniques. For example, one can customize TextPro to provide better segmentation for web data.

Parsing. A model has been trained for Italian on the Torino Treebank data[1] using the Berkeley parser by the Fondazione Bruno Kessler. The treebank being relatively small, a better performance can be achieved by enforcing TextPro part-of-speech tags when training and running the parser. Both the Torino Treebank itself and the parsing model use specific tagsets that do not correspond to the Penn TreeBank tags of the English parser. To facilitate cross-lingual processing and enable unlexicalized cross-lingual modeling for deep semantic tasks, we have mapped these tagsets to each other.

2.2 Coreference pipeline

Once the preprocessing pipeline has created a rich linguistics representation of the input documents, a statistical coreference resolver runs a sequence of sub-modules to provide appropriate information to its model, train/run its classifier and use the output to create coreference chains . This involves the following steps:

- **Mention extraction.** The goal of this step is to extract nominal *mentions* from the textual stream. The exact definition of what is to be considered a mention varies across different annotation schemes. Roughly speaking, nominal chunks, named entities and pronouns (including zeroes) are potential mentions. More fine-grained schemes distinguish between different type of mentions (e.g., referential vs. non-referential) and discard some of them from the scope of their annotation.
- **Mention description.** This component provides a meaningful representation of each

[1] http://www.di.unito.it/~tutreeb/

mention, extracting its linguistic properties, for example: mention type, number, gender and semantic class.

- **Feature extraction.** This component relies on mention descriptions to create feature vectors for the classifier. The exact nature of the feature vector depends on the selection of the underlying model. Thus, in the *mention-pair* model (Soon et al., 2001), used in our study, each vector corresponds to two mentions from the same document, the anaphor and the antecedent. The individual features, engineered manually, combine different bits of information from the corresponding descriptions. An example of such a feature is "the anaphor is a pronoun and it agrees in gender with the antecedent".
- **Modeling.** At the final step, the classifier is trained and tested on the feature vectors and its prediction is then passed to a clustering algorithm to create the resulting partition.

In this paper, we focus on the first two steps, since they require the largest language-specific engineering effort. We believe that the modeling part is relatively language-independent and that most high-performance state-of-the-art models can be applied to Italian if adequate feature representations can be extracted. In our study, we rely on the simple and fast mention-pair model (Soon et al., 2001). We have tested several machine learners (Decision Trees, SVMs and MaxEnt), observing that the highest performance is achieved with decision trees.

3 Experiments

For our experiment, we use a cleaned up version of the LiveMemories Wikipedia corpus (Rodríguez et al., 2010). The first version of the same dataset was adopted for the Anaphora Resolution track at Evalita-2011 (Uryupina and Poesio, 2012). We have invested considerable efforts in checking the consistency of the annotations and adjusting them when necessary. The second version of the corpus will be publicly available by the end of 2014.

The LiveMemories Wikipedia corpus provides rich annotations of nominal mentions. In particular, each mention is characterized for its number, gender, semantic class and referentiality. We will not assess the impact of referentiality on the final performance in this paper, since no automatic referentiality detector has been proposed for Italian so far. However, the corpus does not contain any gold-standard annotations of the basic linguistic levels: all the preprocessing was conducted using automatic modules.

In our experiments, we replace the LiveMemories basic levels with the LiMoSINe pipeline, since it relies on the more recent and robust technology. For coreference components, we start with the oracle pipeline: we extract mentions from the gold annotations and use gold attributes to provide mention descriptions. The performance level of the system with the oracle pipeline can be considered the upper bound for the selected feature extractors and model configurations. The first row of Table 1 summarize the performance level of such a system. We report F-scores for the three most commonly used metrics for coreference (MUC, B^3 and $CEAF_{\phi_3}$). We then gradually replace the oracle components with the automatic ones, measuring the drop in the system's performance.

3.1 Mention Description

In our first experiment, we take gold mention boundaries and try to describe mention properties automatically. To this end, we try to extract the head of each mention. We traverse the parse tree for mentions corresponding to parse nodes. For other mentions, we rely on simple heuristics for extracting head nouns. Once the head noun has been extracted, we consult the TexPro morphology to determine its number and gender. If the mention aligns with some named entity, as extracted by TextPro, we also assign it a semantic type.

This methodology may lead to incomplete or incorrect mention descriptions for various reasons. First, the head-finding rules, especially for mentions that do not correspond to any parsing nodes (this can happen, for example, if the parsing tree is erroneous itself), are not perfect. Second, the TextPro morphology may provide misleading cues. This two types of errors can be remedies in the future with the advancement of the NLP technology. The third group of errors are the cases when the LiveMemories annotators assign some attributes to a mention to agree with other members of its coreference chain. For example, pronouns often receive semantic type attributes that can not be inferred from the corresponding one-sentence contexts. For such cases, a joint model for mention description and coreference resolution might be beneficial. Denis and Baldridge (2009) propose an example of such a model for joint coreference resolution and NE classification.

Components	MUC	CEAF	B^3
Gold boundaries, gold descriptions	50.1	67.8	78.4
Gold boundaries, automatic descriptions	49.2	66.0	77.2
Gold boundaries with no zero pronouns, automatic descriptions	49.5	65.8	76.4
Automatic boundaries, automatic descriptions	44.0	50.3	52.2

Table 1: The system performance with automatic and oracle modules, F-scores.

The second row of Table 1 shows that while the system performance decreases with imperfect mention descriptions, the drop is not large. We believe that this can be explained by two factors:

- unlike many other datasets, the LiveMemories corpus provides two boundaries for each mention: the *minimal* and *maximal* span; since minimal spans are very short and typically contain 1-3 words, the head finding procedure is more robust;
- while the system is not always able to extract implicit properties for pronouns, the explicit morphology (number and gender) often provides enough information for the coreference resolver; this is in a sharp contrast with the same task for English, where the lack of explicit morphological marking on candidate antecedents makes it essential to extract implicit properties as well.

3.2 Mention Extraction

In our second experiment, we replace the mention extraction module with the automatic one. The automatic mention extractor is a rule-base system developed for English and adjusted for Italian (Poesio et al., 2010). Since the system cannot handle zero pronouns, we do the assessment in two steps. For the first run (row 3 in Table 1), we take all gold mentions that are not zeroes and thus provide a more accurate upper bound for our approach. For the second run (row 4), we do the mention extraction fully automatically. Both runs rely on the automatic mention description component and do not use any gold information apart from mention boundaries.

The most surprising results is the performance level of the system with no zero pronouns. When we remove them from the oracle, the performance doesn't decrease at all. This can be explained by the fact that zero pronouns are very different from other types of mentions and require special algorithms for their resolution. The general-purpose system cannot handle them correctly and produces too many errors. We believe that while zero pronouns pose a challenging problem, the more promising approach would treat them separately from other anaphors, capitalizing on various syntactic clues for their identification and resolution. An example of such an approach for Italian has been advocated by Iida and Poesio (2011).

Altogether, when gold mention boundaries are replaced with the automatic ones, the performance goes down considerably. This is a common problem for coreference and has been observed for many other languages. This finding suggests that the first step in boosting the performance level of a coreference resolver should focus on improving the mention extraction part.

4 Conclusion

In this paper, we have attempted an extensive evaluation of the impact of two language-specific components on the performance of a coreference resolver for Italian. We show that the mention extraction module plays a crucial role, while the contribution of the mention description model, while still important, is much less pronounced. This suggests that the mention extraction subtask should be in the primary focus at the beginning of the language-specific research on coreference. Our future work in this direction includes developing a robust statistical mention detector for Italian based on parse trees.

We also show that zero pronouns can not be handled by a general-purpose coreference resolver and should therefore be addressed by a separate system, combining their extraction and resolution.

Finally, our study has not addressed the last language-specific component of the coreference pipeline, the feature extraction module. Its performance cannot be assessed via a comparison with an oracle since there are no perfect gold features. In the future,we plan to evaluate the impact of this component by comparing different feature sets, engineered both manually and automatically.

References

Giuseppe Attardi, Stefano Dei Rossi, and Maria Simi. 2012. UNIPI participation in the Evalita 2011 Anaphora Resolution Task. In Bernardo Magnini, Francesco Cutugno, Mauro Falcone, and Emanuele Pianta, editors, *Lecture Notes in Computer Science 7689*. Springer.

Pascal Denis and Jason Baldridge. 2009. Global joint models for coreference resolution and named entity classification. In *Procesamiento del Lenguaje Natural 42, Barcelona: SEPLN*.

Ryu Iida and Massimo Poesio. 2011. A cross-lingual ILP solution to zero anaphora resolution. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 804–813.

Valentina Bartalesi Lenzi and Rachele Sprugnoli. 2009. EVALITA 2009: Description and results of the local entity detection and recognition (LEDR) task. In *Proceedings of Evalita-2009*.

Emanuele Pianta, Christian Girardi, and Roberto Zanoli. 2008. The TextPro tool suite. In *Proceedings of the Language Resources and Evaluation Conference*.

Massimo Poesio, Olga Uryupina, and Yannick Versley. 2010. Creating a coreference resolution system for Italian. In *Proceedings of the Linguistic Coreference Workshop at the International Conference on Language Resources and Evaluation (LREC'10)*.

Sameer Pradhan, Lance Ramshaw, Mitchell Marcus, Martha Palmer, Ralph Weischedel, and Nianwen Xue. 2011. Conll-2011 shared task: Modeling unrestricted coreference in ontonotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning (CoNLL 2011)*, Portland, Oregon, June.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Proceedings of the Sixteenth Conference on Computational Natural Language Learning (CoNLL 2012)*, Jeju, Korea.

Marta Recasens, Lluís Màrquez, Emili Sapena, M.Antònia Martí, Mariona Taulé, Véronique Hoste, Massimo Poesio, and Yannick Versley. 2010. SemEval-2010 Task 1: Coreference resolution in multiple languages. In *Proceedings of the 5th International Workshop on Semantic Evaluations (SemEval-2010)*, Uppsala, Sweden.

Kepa Joseba Rodríguez, Francesca Delogu, Yannick Versley, Egon Stemle, and Massimo Poesio. 2010. Anaphoric annotation of wikipedia and blogs in the live memories corpus. In *Proceedings of the 7th International Conference on Language Ressources and Evaluation*.

Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistic*, 27(4):521–544.

Olga Uryupina and Massimo Poesio. 2012. EvalIta 2011: Anaphora resolution task. In Bernardo Magnini, Francesco Cutugno, Mauro Falcone, and Emanuele Pianta, editors, *Lecture Notes in Computer Science 7689*. Springer. (extended version).