Creating a Coreference Resolution System for Italian

Massimo Poesio*, **Olga Uryupina***, **Yannick Versley**‡

*CiMeC, University of Trento
‡SFB 833, University of Tübingen
massimo.poesio@unitn.it, uryupina@gmail.com, versley@sfs.uni-tuebingen.de

Abstract

This paper summarizes our work on creating a full-scale coreference resolution (CR) system for Italian, using BART – an open-source modular CR toolkit initially developed for English corpora. We discuss our experiments on language-specific issues of the task. As our evaluation experiments show, a language-agnostic system (designed primarily for English) can achieve a performance level in high forties (MUC F-score) when re-trained and tested on a new language, at least on gold mention boundaries. Compared to this level, we can improve our F-score by around 10% introducing a small number of language-specific changes. This shows that, with a modular coreference resolution platform, such as BART, one can straightforwardly develop a family of robust and reliable systems for various languages. We hope that our experiments will encourage researchers working on coreference in other languages to create their own full-scale coreference resolution systems – as we have mentioned above, at the moment such modules exist only for very few languages other than English.

1. Introduction

This paper presents a coreference resolution system for Italian based on BART (Versley et al., 2008). BART is a modular toolkit for coreference resolution that supports state-of-the-art statistical approaches to the task and enables efficient feature engineering. BART has originally been created and tested for English, but its flexible modular architecture ensures its portability to other languages and domains.

Even though the basic linguistic notions used in coreference resolution – noun phrases, pronouns, definiteness markers – can be found in a relatively wide range of Germanic and Romance languages, relatively few coreference systems aim at covering multiple languages using one coreference component. While there are linguistic differences to English within these languages – Romance languages such as Italian and Spanish have zero subjects and empty pronouns – the greater obstacle seems to be that the tagsets or syntactic structures commonly used for processing these languages differ considerably, but also that processing tools which are easily available for English are not available or difficult to obtain for other languages. In comparison to languages such as Japanese, however, where definiteness is not marked and zero pronouns can also occur in object position, which makes detailed syntactic-semantic information necessary,[1] the variance within Germanic and Romance languages is small enough that it is conceivable that one system could perform coreference resolution on all of these languages.

Several researchers did attempt to port their approach to different languages: (Mitkov et al., 1998), who present adaptations of the MARS approach to pronoun resolution to Polish and Arabic, (Harabagiu and Maiorano, 2000), who perform resolution on English-Romanian parallel texts, and (Luo and Zitouni, 2005), who perform coreference resolution on English, Chinese and Arabic data and compare the usefulness of syntactic features for coreference resolution in these languages.

To our knowledge, we present a first full-scale coreference resolution system for Italian. Note also that the system operates on a raw text and not on a set of predefined "gold" mentions – i.e. it can be used as a module for a real-world application.

We have evaluated our system on the ICab dataset (Magnini et al., 2007). The data comprise articles for four days of the "Adige" newspaper. We have used one day of "Adige" for testing and three days – for training. On this split, our system achieves a performance level of 56.1% (MUC F-score). On a similar task for English, state-of-the-art tools achieve a slightly better performance level (low to mid sixties for system mentions on MUC or ACE corpora). However, an English system retrained on Italian dataset yields much lower performance figures. We believe that our improvement over this baseline is due, in part, to the specialized mention tagger (Biggio et al., 2009) and, in part, to a number of language-specific adjustments to BART discussed below.

2. BART Architecture

The BART coreference resolution toolkit has four main components: preprocessing pipeline, feature extraction module, decoder and encoder. In addition, an independent *LanguagePlugin* module handles all the language specific information and is accessible from any component. The architecture is shown on Figure 1. Each module can be accessed independently and thus adjusted to leverage the system's performance on a particular language or domain.

The preprocessing pipeline converts an input document into a sequence of mentions with assigned properties (number, gender etc). The feature extraction module describes pairs of mentions $\{M_i, M_j\}$, $i < j$ as a set of features. Table 1 shows the features we used. All the feature values are computed automatically, without any manual intervention.

[1] Some earlier experiments on coreference resolution in Japanese, such as (Aone and Bennett, 1995) just assume that zero pronouns are given beforehand, an assumption of questionable practical value.

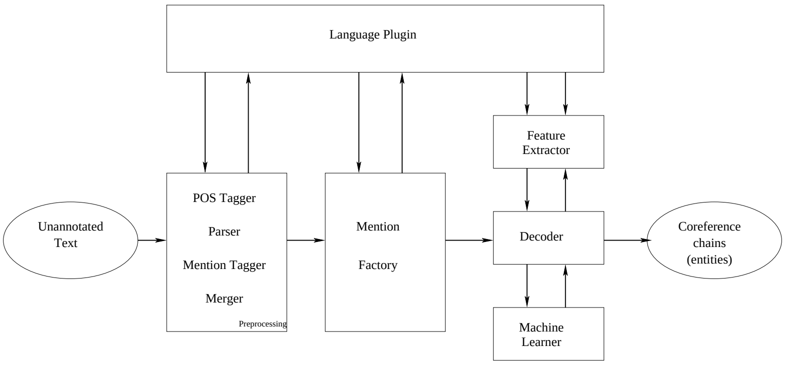

Figure 1: BART architecture

Features
MentionType(M_i)
MentionType(M_j)
SemanticClass(M_i)
SemanticClass(M_j)
FirstMention(M_i)
GenderAgreement(M_i,M_j)
NumberAgreement(M_i,M_j)
AnimacyAgreement(M_i,M_j)
Alias(M_i,M_j)
Apposition(M_i,M_j)
StringMatch(M_i,M_j)
Distance(M_i,M_j)

Table 1: Features used by our Italian version of BART: each feature describes a pair of mentions $\{M_i, M_j\},\ i < j$, where M_i is a candidate antecedent and M_j is a candidate anaphor

	Recall	Precision	F
	MUC		
universal	17.2	79.2	28.3
Italian	22.5	90.7	36.0
	CEAF		
universal	80.6	43.4	56.4
Italian	68.7	49.3	57.4

Table 2: Performance (MUC and CEAF-ϕ_4 recall, precision and F scores) of the *alias* feature

The decoder generates training examples through a process of sample selection and learns a pairwise classifier. Finally, the encoder generates testing examples through a (possibly distinct) process of sample selection, runs the classifier and partitions the mentions into coreference chains given the classifier decisions. For our Italian CR engine, we have tested a number of machine learning algorithms and decoding/encoding techniques and have opted for the setting advocated by (Soon et al., 2001) with the maximum entropy classifier.

3. Developing language-specific components

Our work on adapting BART to Italian has followed two directions: we have developed an Italian language plugin and a new preprocessing pipeline.

Aliasing. Our work on the language plugin has mostly included investigating Italian-specific aliasing techniques. A list of company/person designators (e.g., “S.p.a” or “D.ssa”) has been manually crafted. We have extracted from the training data several patterns of name variants for the locations (e.g. “Provincia di Verona” and “Verona” may refer to the same place). Finally, we have relaxed abbreviation constraints, allowing for lower-case characters in the abbreviations – a pattern that is much more common for Italian than for English. Table 2 shows the performance level for a coreference resolution system based on the aliasing feature alone. The first row represents a language-agnostic approach to aliasing, the second row – the aliasing approach we have created specifically for Italian. It suggests that, although a universal aliasing algorithm is able to resolve some coreference links between named entities, creating a language-specific module boosts the system’s performance substantially. It should be noted that most coreference resolution systems rely on a very generic approach to aliasing, ignoring any language-specific structures of proper names. We believe that a coreference resolution system could benefit a lot from a more sophisticated aliasing algorithm (cf., for example, (Patman and Thompson, 2003) for a related study from the text mining community).

Preprocessing. We have run several evaluation experiments with the different designs of the preprocessing pipeline to optimize the system’s performance on the Icab dataset. For the testing data, the preprocessing is straightforward: we input all the chunks detected by a mention tagger (Biggio et al., 2009) and assign relevant properties from the output of the corresponding component of a shallow NLP toolkit for Italian, TextPro (Pianta et al., 2007). The properties include part-of-speech, morphological features such as number and gender, as well as semantic type. For the training data, however, this strategy leads to only a moderate performance level for two main reasons.

First, manually annotated (“gold”) mentions tend to be much longer than those extracted by the tagger (“system mentions”). This means that our matching and aliasing models, learned directly from the gold training data, may

	Recall	Precision	F
	MUC		
shallow pipeline	45.8	72.3	56.1
parsing pipeline	42.4	73.7	53.8
	CEAF		
shallow pipeline	62.1	64.6	63.3
parsing pipeline	63.8	62.0	62.9

Table 3: System performance (MUC and CEAF-ϕ_4 recall, precision and F scores) with different preprocessing pipelines

	Recall	Precision	F
	MUC		
universal	34.9	76.6	47.9
Italian	46.8	71.1	56.4
	CEAF		
universal	82.4	51.7	63.6
Italian	78.6	57.4	66.3

Table 4: Performance (MUC and CEAF-ϕ_4 recall, precision and F scores) on gold mentions: language agnostic vs. Italian-specific system

not be applicable to automatically extracted testing mentions. To rectify this problem, we have adjusted gold mention boundaries to cover only the heads, not the extents.

Second, the training data contain a number of embedding mentions – chunks that span over another mention (e.g. "la popolazione del sobborgo" is a mention of the second level of embedding, as it spans over another, first-level mention, "sobborgo"). Our mention tagger can only extract mentions of the first and second level of embedding. We have, therefore, discarded all the gold mentions with the higher level of embedding to avoid unnecessary noise.

We have also investigated an alternative parsing pipeline: within this strategy, the chunks, suggested by the mention tagger, are mapped into NP-like nodes in automatically constructed parse trees[2]. The parser consisted on a dependency parser (Nivre et al., 2007) trained on a converted version of the Torino University Treebank (Bosco and Lombardo, 2006), a freely available treebank for Italian, and a dependency-to-constituency converter. Because of the vastly larger size of iCab compared to the Torino treebank (ICab contains about 350.000 tokens of text whereas the treebank only contains about 60.000 tokens), the simpler and faster chunking-based pipeline works much better than full parsing (cf. Table 3). This is in stark contrast to English, where state-of-the-art parsing gives better results than even the best available chunkers.

Note that morphological preprocessing for Italian, on the contrary, is much easier and more accurate, than for English: thus, we can reliably obtain mentions properties (e.g., gender) from a shallow morphological analyzer (TextPro).

	Recall	Precision	F
	MUC		
Italian	45.8	72.3	56.1
	CEAF		
Italian	62.1	64.6	63.3

Table 5: Performance (MUC and CEAF-ϕ_4 recall, precision and F scores) on automatically extracted mentions

4. Evaluation

Our evaluation experiments follow two objectives. First, we want to find out, to what extent a generic language-agnostic system can be used for a new language. Second, we try to estimate the impact of our language-specific adjustments.

For the language-agnostic setting, we have taken the English version of BART, substituted all the external modules (tokenizer, POS-tagger, parser) with the Italian ones and re-trained the system on the Evalita dataset. Unfortunately the results are very moderate mainly due to the annotation guidelines: following the ACE standards, only a subset of mentions has been annotated for coreference, making a data-specific mention tagger a vital part of the system. We have therefore started by comparing our language-agnostic and Italian systems on the gold mentions (Table 4).

In our last experiment we rely on an Italian mention tagger (Biggio et al., 2009) to detect mention boundaries. As this is a crucial part of a coreference-resolution system, we cannot replicate this experiment for the language-agnostic system. Table 5 shows the system performance with the mention tagger (i.e. when operating on a raw text, with no manual intervention).

As our evaluation experiments show, a language-agnostic system (designed primarily for English) can achieve a performance level in high forties (MUC F-score) when retrained and tested on a new language, at least on gold mention boundaries. Though this number might appear low, note that it is a baseline requiring no extra engineering. Compared to this level, we can improve our F-score by around 10% introducing a small number of language-specific changes. This shows that, with a modular coreference resolution platform, such as BART, one can straightforwardly develop a family of robust and reliable systems for various languages. We hope that our experiments will encourage researchers working on coreference in other languages to create their own full-scale coreference resolution systems – as we have mentioned above, at the moment such modules exist only for very few languages other than English.

5. Conclusion

To summarize, we have extended BART (Versley et al., 2008) to create a full-scale coreference resolution system for Italian. Its modular design has allowed us to port a large part of the functionality from English to Italian with no changes – we have only had to run a series of evaluation runs on the development set to pick the best decoding/encoding scheme and the most suitable machine learn-

[2] This pipeline shows reliable performance on English data.

ing algorithm from a range of solutions provided in the BART distribution. We have therefore focused our attention on improving the system's performance by taking care of language-specific properties. Our experiments have shown that a coreference resolution system based on shallow preprocessing works better for a morphologically rich language, such as Italian, compared to parsing-oriented strategies more common for English.

6. References

Chinatsu Aone and Scott Bennett. 1995. Evaluating automated and manual acquisition of anaphora resolution strategies. In *Proc. ACL 1995*.

Silvana Marianela Bernaola Biggio, Claudio Giuliano, Massimo Poesio, Yannick Versley, Olga Uryupina, and Roberto Zanoli. 2009. Local entity detection and recognition task. In *Proceedings of Evalita-2009*.

Cristina Bosco and Vincenzo Lombardo. 2006. Comparing linguistic information in treebank annotations. In *LREC 2006*.

Sanda Harabagiu and Steven Maiorano. 2000. Multilingual coreference resolution. In *Sixth Applied Natural Language Processing Conference (ANLP-NAACL 2000)*.

Xiaoqiang Luo and Imed Zitouni. 2005. Multi-lingual coreference resolution with syntactic features. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*.

Bernardo Magnini, Emanuele Pianta, Manuela Speranze, Valentina Bartalesi Lenzi, and Rachele Sprugnoli. 2007. Italian content annotation bank (i-cab). Technical report, FBK-IRST.

Ruslan Mitkov, Lamia Belguith, and Malgorzata Stýs. 1998. Multilingual robust anaphora resolution. In *EMNLP 1998*.

Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gulsen Eryigit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Frankie Patman and Paul Thompson. 2003. Names: A new frontier in text mining. In *Proceedings of the 1st NSF/NIJ Symposium*, pages 27–38.

Emanuele Pianta, Christian Girardi, and Roberto Zanoli. 2007. The textpro tool suite. Technical report, FBK-IRST.

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

Yannick Versley, Simone Paolo Ponzetto, Massimo Poesio, Vladimir Eidelman, Alan Jern, Jason Smith, Xiaofeng Yang, and Alessandro Moschitti. 2008. BART: A modular toolkit for coreference resolution. In *Proceedings of the Linguistic Coreference Workshop at the International Conference on Language Resources and Evaluation (LREC-2008)*.