Error Analysis for Learning-based Coreference Resolution

Olga Uryupina

Institute of Linguistics,
Russian Academy of Science
B. Kislovsky per. 12/1,
Moscow, Russia
uryupina@gmail.com

Abstract

State-of-the-art coreference resolution engines show similar performance figures (low sixties on the MUC-7 data). Our system with a rich linguistically motivated feature set yields significantly better performance values for a variety of machine learners, but still leaves substantial room for improvement. In this paper we address a relatively unexplored area of coreference resolution – we present a detailed error analysis in order to understand the issues raised by corpus-based approaches to coreference resolution.

1. Introduction

Robust coreference resolution is essential for various NLP tasks, such as Information Extraction or Question Answering. Although there has been much attention to the problem, state-of-the-art coreference resolution algorithms still only have a moderate performance (around 60% F-measure for coreference chains on the MUC-7 data).

Recent studies (Cristea et al., 2002; Barbu et al., 2002) claim that existent (knowledge-poor) algorithms are only able to account for "easy" coreference links and suggest more sophisticated frameworks to deal with complex anaphora resolution cases.

We have built a learning-based coreference resolution engine incorporating various kinds of linguistic information (Uryupina, 2006; Uryupina, 2007). Our system relies on 351 nominal features (1096 boolean/continuous), representing surface (122 features), syntactic (64), semantic (29), and salience-based (136) properties of markables and markable pairs. We have evaluated its performance for a variety of publicly available machine learners (SVM^{light}, C4.5, Ripper, Slipper, MaxEnt), observing a consistent significant improvement over the state-of-the-art level (Soon et al., 2001). The system's performance with the SVM^{light} classifier (F-score of 65.4%) is, to our knowledge, the best result on the MUC-7 data reported so far in the literature. Moreover, our learning curves (see Figure 1, dashed line) show no signs of convergence, suggesting that we can get even better performance values by annotating more training material or improving our learning strategy.

Although our rich feature set has brought significant improvement, the results still lie in sixties. Our SVM^{light} classifier has missed 469 of the 1299 manually annotated links (recall value of 63.9%) and suggested 408 spurious links (precision value of 67.0%).

We have performed a detailed error analysis to identify possible directions for future work on our system. We also believe that our error analysis would be helpful for any coreference resolution algorithm: most state-of-the-art systems share very similar performance figures on the MUC data and we therefore assume that they also share very similar problems.

We discuss the most common errors in Section 2. The error analysis has revealed several major problems with our approach (and, we believe, most state-of-the-art coreference resolution systems): insufficient data quality (Section 3.), shortages of preprocessing modules (Section 4.), inadequate features (Section 5.) and deficiencies in the resolution strategy (Section 6.).

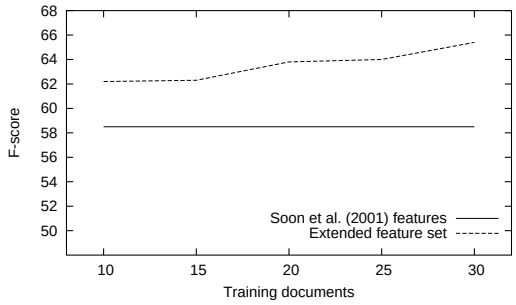

Figure 1: Learning curve (F-score) for SVM^{light}: linguistically rich feature set (dashed line) vs. Soon et al. (2001) features (solid line).

2. Distribution of errors

Below we provide a very brief overview of the errors made by our system. Tables 1 and 2 summarize the distribution of our recall and precision errors. Additional examples and a more detailed analysis are provided in (Uryupina, 2007).

2.1. Recall errors

The most common types of recall errors are: missing markables (35%), deficiencies in nominal anaphora resolution (22%), and incorrectly (un-)resolved pronouns (16%).

The most crucial issue is clearly markables' extraction. The MUC-7 annotation guidelines consider as a *markable* any NP or NP-like unit participating in a coreference chain. The pool of markables, as annotated in the MUC-7 corpus, is therefore very heterogeneous. For example, the guidelines instruct the annotators to analyze not only the text-body of a document, but also its auxiliary parts (SLUG,

DATE, NWORDS, PREAMBLE, TRAILER)[1]:

<SLUG fv=taf-z> BC-[LORAL-SPACE]-470&ADD-N </SLUG>

Extracting markables from such semi-structured data is a non-trivial task of its own. As Table 1 suggests, around one third of the missed markables come from auxiliary document parts, even though they are on average very short.

For the textbody, state-of-the-art coreference resolution systems rely on external automatic modules for computing the set of markables and therefore some anaphors and antecedents are inevitably missed. Even the most sophisticated coreference resolution algorithm cannot, obviously, account for a coreference link if the anaphor or its antecedent(s) are not recognized. Unfortunately, there are virtually no studies on improving the interaction of coreference resolution engines with their preprocessing modules. We will come back to this issue in Section 4.

Another big problematic area is nominal coreference – complex anaphora resolution cases, involving sophisticated inference schemes (for example, linking together in a chain "satellites", "U.S. reconnaissance technology", "advance intelligence-gathering tools" and "remote-sensing instruments") in the following example:

As peaceful as that may seem, a report on the $[satellites]_1$*' findings, completed in March, was designated as secret because the information could reveal too much about the abilities of* $[U.S.\ reconnaissance\ technology]_{2,ante=1}$*... Rather, the study's importance lay in the use of* $[advanced\ intelligence\text{-}gathering\ tools]_{3,ante=2}$ *to examine the environment, an application that scientists say has enormous potential benefits for future research.* $[Remote\text{-}sensing\ instruments]_{4,ante=3}$ *could save time and money in various projects, producing data that would otherwise be hard to gather.*

Although we have a number of WordNet-based features to account for semantic compatibility, the system still achieves only moderate results in resolving such links. Nominal anaphora is a well-known difficult problem, discussed in the literature.

Complex cases of pronominal anaphora account for 16% of our recall errors. Salience-based intersentential pronoun resolution is relatively reliable, but same-sentence pronominal anaphora remains problematic:

"The $[cable\ operator]_1$ *doesn't care how old* $[[his]_{3,ante=1}\ subscriber]_2$ *is as long as* $[he]_{4,ante=2,\neq 1}$ *pays* $[his]_{5,ante=2,\neq 1}$ *monthly bill."*

Another rather unaddressed issue is the resolution of 1st and 2nd person pronouns – most state of the art coreference resolution algorithms rely on no specific techniques for dealing with "I", "we", or "you". These pronouns are potentially problematic for any system, because their resolution often involves complex discourse modeling:

The retiring Republican chairman of the House Committee on Science wants $[U.S.]_1$ *businesses to compete in the commercial launch industry... (9 sentences) "*$[We]_{2,ante=1}$ *need to make it easier for the private sector to compete in the space industry," Walker said.*

Some syntactic constructions (apposition and copula) are very strong indicators for coreference, but they can often be confused with other syntactic structures and therefore require sophisticated extraction patterns, based on a parser's output. We identify candidates for appositions and copulas with a regular expression matcher, and then refine the candidate set, discarding, for example, addresses or coordinate constructions. This procedure relies crucially on the parser's quality. Appositions, however, are intrinsically difficult for parsing (see Section 4. below), leading to incorrect values for syntactic features and decreasing the system's performance.

Around 7% of our recall errors are matching mistakes: the classifier fails to link two variants of the same name. Names of ORGANIZATION are the most difficult NE-anaphors for our system, contributing to 20 of 31 matching-related recall errors. Organizations are typically introduced by their official names and then further re-mentioned by simplified descriptions: some words can be abbreviated, and some omitted. Coreference links between a full and an abbreviated version of the same name are under-represented in our training corpus and therefore the learners cannot reliably extract them. We expect to get better results on abbreviations by adding more training material.

Some rather infrequent name-matching patterns are still not covered by our features. For example, we can match "Mild Seven Benetton Renault F1 Team" to "Benetton" or "Renault", but we fail to link "Ziff-Davies Publishing Co" to "Ziff", as we do not treat the hyphenation mark as a word separator.

Finally, propagated precision errors also decrease our recall. This is a problem of all the coreference resolution algorithms within the framework of Soon et al. (2001). We discuss it in detail in Section 6. below.

2.2. Precision errors

The most common types of precision errors are: deficiencies in nominal anaphora resolution (45%), incorrectly resolved pronouns (19%), and misleading markables (19%).

Around half of the precision errors made by our system are incorrectly resolved full noun phrases. Our classifier mainly relies on the family of `same_head` features for nominal anaphora resolution.

It is generally assumed in the literature that one should pay closer attention to (pre-) modifiers to determine whether two same-head NPs are coreferent: for example, "the state-owned French companies" and "U.S. companies" below can hardly refer to the same object, because "French" and

[1] In all the examples throughout this paper markables are shown with square brackets.

"U.S." are incompatible:

While [*the state-owned French companies'*]$_1$ *rivals across the Atlantic have been "extremely impressive and fast" about coming together in mergers,* [*European companies*]$_{2,ante\neq1}$*, hobbled by political squabbling and red tape, have lagged behind, Gallois said... The competition is even tougher for Aerospatiale in that the U.S. dollar has weakened 10 percent against the French franc last year, giving* [*U.S. companies*]$_{3,ante\neq1,\neq2}$ *what Gallois called a "superficial" advantage.*

The bottleneck of this approach lies in the lack of required knowledge bases: we can compile small lists of mutually incompatible properties, but a large-scale general-purpose resource can hardly be produced manually in any reasonable time. In addition, some properties are generally compatible, but can become incompatible in specific contexts (consider "European companies" and "French companies" in the example above).

The distribution of precision errors clearly shows that in most cases we cannot even potentially rely on modifiers' incompatibility: often the anaphor and its candidate antecedent have compatible modifiers or at least one of them is not modified at all. In such cases we need a deeper analysis, involving multiple linguistic factors.

At least some spurious links between same-head noun phrases can be eliminated by discarding discourse-new markables (see (Poesio et al., 2004) for an overview of relevant algorithms). If a candidate anaphor is likely to be a discourse new entity, the link is highly implausible:

If you have a ship that can fire Tomahawk missiles, and fire anti-air missiles, and maybe fire ATACMS (Army Tactical Missiles), [*that ship*]$_1$ *will perform a function that* [*some other ship*]$_{2,ante\neq1}$ *won't have to perform.*

For pronominal anaphora, we have identified two major subclasses of precision errors: overestimating the impact of matching features (resolving, for example, "it" to "it") and incorrect preference for salient same-sentence candidates. Comparing this to the distribution of our recall errors, we can conclude that our system performs only moderately on same-sentence pronominal anaphora. This is not surprising, because our approach, following most state-of-the-art pronoun resolution systems, relies a lot on salience-based features.

Deficiencies in markables' extraction and spurious values for syntactic features decrease the system's precision in the similar way as they affect its recall (see Section 2.1. above).

Finally, around 5% of the spurious links are incorrectly matched named entities. Newswire documents often describe distinct entities with similar names – relatives (PERSON) or spin-off companies (ORGANIZATION). The snippet below mentions "Loral", "Loral Space and Communications Corp.", "Loral Space", and "Space Systems Loral":

News of Monday's deal, in which Lockheed will buy most of [*Loral*]$_1$*'s military businesses and invest $344 million in* [*Loral Space and Communications Corp.*]$_2$*,* [*a new company*]$_{3,ante=2}$ *whose principal holding will be* [*Loral*]$_{4,ante=1}$*'s interest in Globalstar, sent Globalstar's own shares soaring $6.375, to $40.50 in Nasdaq trading... In addition, Schwartz said* [*Loral Space*]$_5$ *would use its holdings in* [*Space Systems Loral*]$_{6,ante=4}$*,* [*a private maker*]$_{7,ante=6}$ *of satellites, to expand into the direct broadcast satellite business.*

It is difficult even for a human reader to correctly cluster these names: "Loral", "Loral Space and Communications Corp.", and "Space Systems Loral" are different companies, whereas "Loral Space" is another name for "Loral Space and Communications Corp".

To summarize, our error analysis shows that some coreference links are intrinsically difficult and can only be accounted for by deep analysis. These are complex anaphora cases, mentioned by Cristea et al. (2002) and (Barbu et al., 2002), including, for example, nominal anaphora or tricky 1st and 2nd person pronouns. A lot of coreference links, however, can still be potentially established by shallow algorithms. In the following sections we discuss possible directions for improvement, starting from the most feasible steps and getting to more complex extensions.

	Errors	%
MUC-7 inconsistencies	17	3.6%
Missing markables	166	35.4%
auxiliary doc parts	50	10.7%
tokenization	8	1.7%
one-word modifiers	36	7.7%
multi-word modifiers	10	2.1%
bracketing/labelling	54	11.5%
other	8	1.7%
Propagated P-errors	31	6.6%
PRO-anaphora	17	3.6%
NP-anaphora	14	3.0%
Pronominal anaphora	77	16.4%
NE-matching	31	6.6%
Syntactic constructions	39	8.3%
apposition	18	3.8%
copula	8	1.7%
quantitative	13	2.8%
NP-anaphora	104	22.2%
same head	4	0.9%
morph. variants	7	1.5%
head-modifier	10	2.1%
NP_{ana}-NP_{ante}	46	9.8%
NP_{ana}-NE_{ante}	28	6.0%
NE_{ana}-NP_{ante}	7	1.5%
NE_{ana}-NE_{ante}	6	1.3%
total	469	100%

Table 1: Recall errors on the testing data (20 MUC-7 "formal test" documents).

	Errors	%
MUC-7 inconsistencies	30	7.4%
Spurious markables	76	18.6%
preamble	24	5.9%
text body	52	12.7%
Pronoun resolution	78	19.1%
NE-matching	20	4.9%
Syntactic constructions	22	5.4%
apposition	12	2.9%
copula	10	2.5%
NP-anaphora	182	44.6%
multi-word expressions	3	0.7%
homonymy	4	1.0%
new modifier (anaphor)	15	3.7%
incompatible modifiers	30	7.4%
compatible modifiers	58	14.2%
no modifiers	62	15.2%
total	408	100%

Table 2: Precision errors on the testing data (20 MUC-7 "formal test" documents).

3. Data

We have used the MUC-7 corpus in our study. It consists of 30 training ("dry-run") and 20 testing ("formal") one-page documents. Below we outline several problems with the theoretic assumptions of the MUC guidelines and the annotation quality.

The definition of IDENT coreference, as advocated by the MUC-7 guidelines, is problematic. Van Deemter and Kibble (2001) point out that the MUC annotation scheme fails to separate the coreference relation proper from several other phenomena, such as bound anaphora or predicate nominals. It is difficult for any classifier to learn such a complex distribution, involving different (though related) phenomena.

The MUC evaluation metric is too biased towards recall:

> *BC-CD-RADIO-*$[$*SHARES*$]_1$*-BLOOM...*
> *CD Radio stock rose 2 7/8 to 13 5/8 in trading of* $[$*400,300 shares*$]_2$*, more than quadruple the three-month daily average of* $[$*88,700 shares*$]_3$*.*

Most coreference resolution systems would link all the three markables into a chain: {"SHARES", "400,300 shares", "88,700 shares"}. But even if some system is able to rule out the possibility of "400,300 shares"and "88,700 shares" referring to the same object, it would have to decide which markable should be kept in the chain, and which one not. The fact that a system has correctly avoided a spurious link is not directly rewarded by the MUC-7 scorer.

Even a substantial improvement in the system's precision (for example, by discarding automatically identified discourse new entities) does not necessary lead to a better MUC F-score. If we want to use a coreference resolution engine as a preprocessing module for some other engine, for example, an Information Extraction system, we might want to have a classifier with a high precision level and therefore opt for another scoring scheme, such as the BCUBED metric (Baldwin et al., 1997).

The corpus is very small and simply does not contain enough material for training (the "formal training" documents provided by MUC-7 are not annotated). Our classifiers show no signs of convergence when we train them on 10, 15, 20, 25, or all the 30 "dry-run" documents (Figure 1). We need a larger dataset (for example, the ACE corpus) to make better use of our rich feature set.

The annotation quality can be improved. Deficiencies of manual annotation for the testing corpus inevitably decrease the evaluation *score* for any rule- or learning-based system. The same problems with the training material make the data noisy and thus potentially deteriorate the *performance level* of any learning-based approach.

To summarize, we have to revise the definition of coreference and the scoring scheme and then accurately annotate more training material. As a first step in this direction, we plan to re-train our classifier on an already existing larger corpus (ACE).

4. Preprocessing modules

We rely on external modules for segmenting MUC documents into sentences[2] (Reynar and Ratnaparkhi, 1997), parsing (Charniak, 2000), NE-tagging (Curran and Clark, 2003) and determining semantic properties of our markables (Miller, 1990). The first three modules are fully automatic corpus-based NLP systems. The WordNet ontology is a large manually created resource.

All the modules have some shortages that may decrease the performance of our system. For example, appositive-coordinate constructions are intrinsically difficult for parsing:

> *Those materials, in turn, were encased in* $[$*Kevlar*$]_1$*,* $[$*a synthetic fiber*$]_{2,ante=1}$*, and* $[$*Nomex*$]_3$ *to achieve a test strength of 400 pounds.*

A typical state-of-the-art parser has no knowledge that helps prefer the 2-entities interpretation ("[[Kevlar], [a synthetic fiber],] and [Nomex]") over the 3-entities interpretation ("[Kevlar], [a synthetic fiber], and [Nomex]").

Errors committed by the external modules result in incorrect markables and spurious or missing links. A possible remedy would be creating a family of mini-parsers, specifically trained to analyze problematic constructions relevant for coreference resolution.

We have to improve the interaction with our preprocessing modules and adjust the external resources to cover specific problems relevant for our task (e.g., train a mini-parser for appositive-coordinate constructions or an NE-tagger for TITLEs or PRODUCTs).

5. Features

Our classifier relies on 351 feature (1096 boolean/continuous). Not all of them, however, are equally important. We have not performed any feature selection and therefore our feature set is highly redundant. Ng and Cardie (2002) have demonstrated that (manual) feature selection can significantly improve the performance

[2] We have not encountered any errors directly caused by incorrect sentence segmentation.

level of a linguistically motivated coreference resolution algorithm.

Some phenomena are covered by many features simultaneously. For example, most of our name-matching and some salience features are produced by enumerating and combining possible values for a set of parameters. This results in a pool of highly inter-correlated features. Even though each feature brings an important bit of information, the whole set has a degree of redundancy that is too high for machine learning. We have to reduce the number of features to get a better classifier.

Some phenomena are covered by our feature set, but the corresponding features are almost ignored by the classifier. For example, we have features to account for abbreviations, but neither C4.5, nor SVM^{light} make any use of them. Our training data does not contain enough abbreviations to learn any reliable patterns. We have to increase the training corpus to get better results.

Finally, some phenomena are still not covered by our feature set. For example, we do not account for quantitative constructions (such as the link between "three-month daily average" and "88,700 shares" in our CD Radio example above). On the one hand, there is always room for improvement: even a system with millions of features can always be augmented with some new information. On the other hand, obtaining values for more sophisticated features is a very difficult task: we need additional external resources and they are likely to introduce errors. We believe that our system already has a lot of encoded information and therefore we have to improve the algorithm itself rather than introduce more knowledge. This view is supported by our learning curves: they show no signs of convergence, suggesting that we still can get better results with the same feature set.

We plan to investigate feature selection and ensemble learning with different feature splits to make better use of our features.

6. Resolution strategy

Our system has a very simple resolution scheme, suggested by Soon et al. (2001) and then followed by most studies on coreference: candidate antecedents for each anaphor are proposed to the classifier one-by-one from right to left until a positive instance is found. The strategy is very local and does not take into account any other markables, when establishing a link between an anaphor and a candidate antecedent.

This may lead to error propagation:

> *The company also said the Marine Corps has begun testing* $[$*two of its radars*$]_1$ *as part of a short-range ballistic missile defense program. That testing could lead to an order for the* $[$*radars*$]_{2,ante=1}$ *that could be worth between $60 million and $70 million.*

Our preprocessing modules have suggested several candidate antecedents for "the radars": "The company", "the Marine Corps", "two", "its radars"... "order". The candidates have been submitted to the classifier one-by-one, starting from the closest markable ("order") and proceeding backwards. The classifier has correctly discarded most candidates, but then at some point has established a spurious link from "the radars" to "its radars". It has never seen any earlier markables, including the correct antecedent "two of its radars".

If our classifier has suggested a spurious antecedent for some markable at an early processing stage (precision error), it will never see any truly positive testing instances and will be unable to resolve the anaphor (recall error). If the classifier has missed the correct antecedent (recall error), it starts processing too distant markables and is likely to suggest some spurious markable (precision error).

Our system sometimes merges several chains into one – it finds pairs of markables (belonging to different chains in the manually annotated data) that seem to be coreferent and links them. The properties of other markables from the affected chains are completely ignored.

Both problems could be avoided by shifting to a more global resolution strategy, operating on chains instead of markables. Theoretic studies of coreference usually have a global view, talking, for example, about "discourse entities". Practical approaches, however, almost never go beyond the markable level. The only algorithm operating directly on chains has been advocated by Luo et al. (2004).

7. Conclusion

In this paper we have presented the error analysis for a data-driven coreference resolution engine.

Our system relies on a rich feature set and is *potentially* able to resolve most difficult anaphora cases discussed in the literature. It yields significantly better performance on the MUC-7 than state-of-the-art systems for a variety of machine learners.

Still, the performance can be improved further. We have performed a detailed analysis of our recall and precision errors (Tables 1 and 2). The data show that some errors can hardly be avoided within a shallow framework (corresponding to "tricky anaphors" of Cristea et al. (2002)). A large group of anaphors, however, could have been resolved correctly even by a corpus-based algorithm, without relying on deeper analysis.

Our error analysis has suggested several directions for future work: improving the training material, more elaborated integration of the external modules, and investigating more global resolution strategies (reasoning in terms of *chains* instead of *markables*).

8. References

Breck Baldwin, Tom Morton, Amit Bagga, Jason Baldridge, Raman Chandraseker, Alexis Dimitriadis, Kieran Snyder, and Magdalena Wolska. 1997. Description of the UPenn camp system as used for coreference. In *Message Understanding Conference Proceedings*.

Catalina Barbu, Richard Evans, and Ruslan Mitkov. 2002. A corpus based investigation of morphological disagreement in anaphoric relations. In *Proceedings of the Language Resources and Evaluation Conference*.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the 1st Meeting of the North*

American Chapter of the Association for Computational Linguistics, pages 132–139.

Dan Cristea, Oana Postolache, and Ruslan Mitkov. 2002. Handling complex anaphora resolution cases. In *Proceedings of the 4th Discourse Anaphora and Anaphor Resolution Colloquium*.

James R. Curran and Stephen Clark. 2003. Language independent NER using a maximum entropy tagger. In *Proceedings of the Seventh Conference on Natural Language Learning*, pages 164–167.

Xiaoqiang Luo, Abe Ittycheriah, Hongyan Jing, Nanda Kambhatla, and Salim Roukos. 2004. A mention-synchronous coreference resolution algorithm based on the bell tree. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*.

George Miller. 1990. Wordnet: An on-line lexical database. *International Journal of Lexicography*.

Vincent Ng and Claire Cardie. 2002. Improving machine learning approaches to coreference resolution. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 104–111.

Massimo Poesio, Olga Uryupina, Renata Vieira, Mijail Alexandrov-Kabadjov, and Rodrigo Goulart. 2004. Discourse-new detectors for definite description resolution: a survey and preliminary proposal. In *Proceedings of the Reference Resolution Workshop at ACL'04*.

Jeffrey C. Reynar and Adwait Ratnaparkhi. 1997. A maximum entropy approach to identifying sentence boundaries. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*.

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 Anaphora Resolution)*, 27(4):521–544.

Olga Uryupina. 2006. Coreference resolution with and without linguistic knowledge. In *Proceedings of the Language Resources and Evaluation Conference*.

Olga Uryupina. 2007. *Knowledge Acquisition for Coreference Resolution*. Ph.D. thesis, Saarland University.