The Dissertation Committee for Pascal Denis

certifies that this is the approved version of the following dissertation:

New Learning Models For Robust Reference Resolution

Committee:

Jason Baldridge, Supervisor

Nicholas Asher, Supervisor

Laurence Danlos

Andrew Kehler

Jonas Kuhn

Raymond Mooney

New Learning Models For Robust Reference Resolution

by

Pascal Denis, B.A., M.Sc.

Dissertation

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Doctor of Philosophy

The University of Texas at Austin

December 2007

To the memory of my father Fernand Denis (1943-1986)

and of my friend Robert L. Dawson (1943-2007)

Acknowledgments

Very few graduate students could truthfully say they had a great adviser: I was fortunate enough to have two. Jason Baldridge and Nicholas Asher have each in their own way provided all a student can ask for: intellectual challenge and moral support. They will remain constant sources of inspiration, both professionally and personally.

Although he joined UT when I was already at an advanced stage of my graduate career, Jason certainly had the strongest impact on it. At a time where I was still all over the place, he helped me, through luminous insights, constant questioning and warm encouragements, channel and structure my ideas. Not the least of things, he managed to make me believe in them. Most of the work reported in this dissertation owes a lot to him, since it grew out of our collaborations and discussions. My regular meetings with Jason were always a cause for rejoicing, for I knew I would leave them smarter and filled up with energy and confidence. I finally owe Jason the special honor of being his first academic offspring.

Nicholas was no less than the reason I came to UT Austin. His depth and breath of knowledge are both impressive and inspiring. His classes and seminars were as intellectually challenging as they were entertaining, and they played a decisive role in my training. Nicholas was always available, and his inputs insightful even when the questions were outside his main area of expertise. I also thank him for his encouragements and his swift responses (while he was in France) during the final stages of the writing.

Next, I wish to thank the members of my committee, Laurence Danlos, Andrew Kehler, Jonas Kuhn, and Ray Mooney for their useful comments and suggestions. A special

thanks goes to Jonas, who actually got me started on this dissertation and was for two years a great mentor and friend before he left UT.

I also owe an important debt of gratitude to other (current or former) linguistics faculty for making the department such a stimulating place. I want to thank in particular: David Beaver, Rajesh Bhatt, Katrin Erk, Bob Harms, Ian Hancock, Richard Meier, Carlota Smith, and Steve Wechsler. I also extend my sincere gratitude to my fellow graduate students: Nicholas Bacuez, Lynda Boudreault, Emmy Destruel, Nick Gaylord, Bernice Hecker, Steve Hilderbrand, Fred Hoyt, Julie Hunter, Eric McCready, Alexis Palmer, Elias Ponvert, Brian Reese, Stéphanie Villard, and Jessica White. I want to single out Brian for being my main resource for obscure idioms and "Americanisms," a common admirer of Seinfeld, PhD Comics, and squirrels, and above all a good friend. Many thanks also to the various members of the UT Natural Language Acquisition Group, in particular Ray Mooney, Razvan Bunescu, Rohit Kate, and John Wong. Thanks finally to Peter Stone and Razvan for their help with CPLEX.

During my graduate training at UT, I was fortunate to spend extended periods of time in various other research institutions: the Lattice group at Université Paris 7, the Institut für Maschinelle Sprachverbarbeitung in Stuttgart, and the Institut de Recherches Informatiques de Toulouse. I want to thank a number of people from these places for making my stay worthwhile and enjoyable, in particular: Pascal Amsili, Nicholas Asher, Laurence Danlos, Martin Forst, Christian Hying, Hans Kamp, Jonas Kuhn, Jerôme Lang, Philippe Muller, Laurent Roussarie, and Jasmin Sarič.

While working on my dissertation, I have received support from different institutional sources for which I am most grateful: the University of Texas Graduate School, the National Science Foundation (grant IIS-0535154), and the Deutsche Akademische Austausch Dienst.

It is hard to imagine working on a dissertation without blowing off steam every now and again. For this reason, I thank my racket partners (whether at squash or tennis),

Jason Baldridge, David Beaver, Bert Meisenbach, Richard Pelton, Joel Sherzer, and Harvey Sussman for letting me beat them on a regular basis; my rock climbing master Nicholas Asher; my smoking partners: Brian, Eric, Esmeralda, and Nicholas; and my road trip buddy Chay Baker. Thanks also to the following friends who managed to stay close despite the distance: Bob, Christian and Julie, Fred, Gilles, Gringo and Tatar, Inti, Louis and Marina, Madeleine and Christian, Myriam and Daniel, Luc and Stéphanie, Pascal and Claire, Phil, Nico.

Finally, I want to thank Sabrina for her love, care, and patience during the last 10 years.

PASCAL DENIS

The University of Texas at Austin

December 2007

New Learning Models For Robust Reference Resolution

Publication No. _____

Pascal Denis, Ph.D.

The University of Texas at Austin, 2007

Supervisors: Jason Baldridge and Nicholas Asher

An important challenge for the automatic understanding of natural language texts is the correct computation of the discourse entities that are mentioned therein —persons, locations, abstract objects, and so on. The problem of mapping linguistic expressions into these underlying entities is known as reference resolution. Recent years of research in computational reference resolution have seen the emergence of machine learning approaches, which are much more robust and better performing than their rule-based predecessors. Unfortunately, perfect performance are still out of reach for these systems. Broadly defined, the aim of this dissertation is to improve on these existing systems by exploring more advanced machine learning models, which are: (i) able to more adequately encode the structure of the problem, and (ii) allow a better use of the information sources that are given to the system.

Starting with the sub-task of anaphora resolution, we propose to model this task

as a ranking problem and no longer as a classification problem (as is done in existing systems). A ranker offers a potentially better way to model this task by directly including the comparison between antecedent candidates as part of its training criterion. We find that the ranker delivers significant performance improvements over classification-based systems, and is also computationally more attractive in terms of training time and learning rate than its rivals.

The ranking approach is then extended to the larger problem of coreference resolution. To main goal is to see whether the better antecedent selection capabilities offered by the ranking approach can also benefit in the larger coreference resolution task. The extension is two-fold. First, we design various specialized ranker models for different types referential expressions (e.g., pronouns, definite descriptions, proper names). Besides its linguistic appeal, this division of labor has also the potential of learning better model parameters. Second, we augment these rankers with a model that determines the discourse status of mentions and that is used to filter the "non-anaphoric" mentions. As shown by various experiments, this combined strategy results in significant performance improvements over the single-model, classification-based approach on the three main coreference metrics: the standard MUC metric, but also the more representative B^3 and CEAF metrics.

Finally, we show how the task of coreference resolution can be recast as a linear optimization problem. In particular, we use the framework of Integer Linear Programming (ILP) to: (i) combine the predictions of three local models (namely, a standard pairwise coreference classifier, a discourse status classifier, and a named entity classifier) in a joint, global inference, and (ii) integrate various other global constraints (such as transitivity constraints) to better capture the dependencies between coreference decisions. Tested on the ACE datasets, our ILP formulations deliver significant f-score improvements over both a standard pairwise model, and various models that employ the discourse status and a named entity classifiers in a cascade. These improvements were again found to hold across the three different evalution metrics: MUC, B^3, and CEAF. The fact that B^3 and CEAF scores

were also improved is of particular importance, since these two metrics are much less lenient than MUC in terms of precision errors.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

An important requisite for the understanding of natural language texts is the correct computation of the *discourse entities* that are mentioned therein —persons, locations, abstract objects, and so on. The problem of mapping linguistic expressions into these underlying entities (irrespective of whether these are seen as real-world objects or intermediate conceptual constructs) is known as **reference resolution**.[1] Computational methods for reference resolution have been developed in Natural Language Processing (NLP) almost since the inception of the field (earlier treatments include Webber (1978) and Hirst (1981) *inter alia*) and they still constitute an area of active research (see (Mitkov, 2002b) for a recent monograph). These approaches have concentrated on two distinct, although closely related, instantiations of this general problem: namely, anaphora resolution and coreference resolution. These two tasks are presented in more detail in Section 1.1. As discussed in Section 1.2, automatic reference resolution is an extremely challenging problem —it is in fact often considered an "AI-complete" problem— and a crucial one —it is key for various other NLP applications. Like in many other areas of NLP, the last decade of research in reference resolution has seen an important shift from hand-crafted systems to machine learning systems. The appli-

[1] The exact nature of these entities is indeed still a matter of debate among philosophers of language. Although this has no direct bearing on the the present work, we assume that discourse entities are conceptual objects (e.g., they can be thought of as first-order logic variables as in Kamp and Reyle (1993)).

cation of standard classification techniques to the tasks of anaphora/coreference resolution has resulted in drastic improvements in robustness, making it theoretically possible to integrate these systems into larger NLP systems (Section 1.3). Performance has also improved, but perfect scores are still out of reach. Broadly defined, the aim of this dissertation is to improve on these existing systems by exploring more advanced machine learning models, which are: (i) able to more adequately encode the structure of the problem, and (ii) allow a better use of the information sources that are given to the system. The goals and contributions of this dissertation are presented in more detail in Section 1.4, and a general outline of the dissertation follows in Section 1.5.

1.1 The different tasks

1.1.1 Anaphora resolution

The first and most studied instantiation of the general problem of reference resolution is **anaphora resolution**. In its broadest sense, anaphora describes an *asymmetric* relation between two linguistic expressions, an **antecedent** and an **anaphor**, wherein the anaphor cannot be fully interpreted without making use of the antecedent. This definition is rather vague and potentially encompasses many different linguistic pheonomena. Consequently, most computational approaches have often assumed a much stricter definition, where: (i) *identity* of reference is required between the anaphor and the antecedent (i.e., they point to the same entity), (ii) both of these expressions are *nominal expressions*, and (iii) only a subset of anaphoric expressions (typically, pronominal ones) are considered.[2] This focus on pronouns is not surprising. Referential pronouns are in a sense the prototypical forms of anaphor: they have no intrinsic semantic content (except gender and number), which makes their interpretation entirely dependent on their antecedent.

The process of anaphora resolution thus defined is shown on the following excerpt

[2] Anaphora can indeed occur between various types of expressions and involves different semantic relations (Clark, 1975; Partee, 1984; Asher, 1993).

from the Automatic Content Extraction (ACE) program[3] corpus in Figure 1.1.

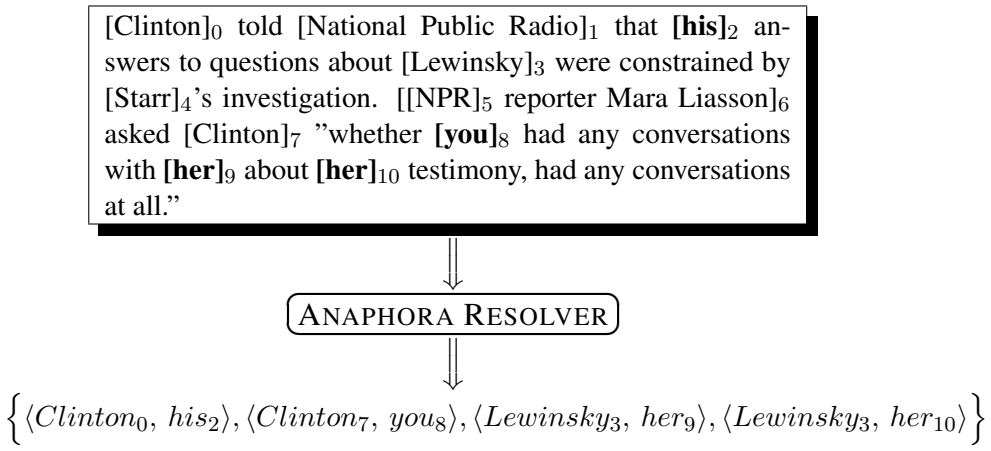

Figure 1.1: The task of anaphora resolution

As illustrated above, an anaphora resolver takes a set of anaphoric expressions as input and outputs an antecedent for each of these anaphors. Note that this description leaves out the question of how the anaphoric expressions are first detected. Typically, anaphora resolution systems assume that the anaphors have already been detected. Note that this is often not a important issue with pronouns, since most uses of pronouns are anaphoric (with the exception of pleonastic uses, for instance), but distinguishing anaphoric and non-anaphoric uses of other types of expressions (e.g., definite descriptions) is a real challenge.[4]

1.1.2 Coreference resolution

Although most computational approaches have focused on anaphora resolution, recent approaches have shifted their attention to the more challenging task of **coreference resolution**. Coreference, also sometimes called *co-specification* (Sidner, 1983), describes the relation that holds between two expressions that refer to the same entity: these are often called **mentions** of that entity. By contrast with anaphora, coreference is an *equivalence* relation:

[3] `http://www.nist.gov/speech/tests/ace/`

[4] In their corpus study, (Vieira and Poesio, 2000) report that more than 50% uses of definite descriptions are discourse-new (i.e., non-anaphoric).

it is reflexive, symmetric, and transitive. Another important difference is that coreference doesn't imply *context-sensitivity* (van Deemter and Kibble, 2000). Two expressions like *George W. Bush* and *Barbara Bush's son* corefer without either of them depending on the other for its interpretation. A correlate of this is that coreference is not "discourse bound", in the sense that coreference can hold across documents. In this dissertation, we will only be concerned by the problem of coreference resolution at the document level. [5]

The process of coreference resolution is illustrated on the same sample excerpt from ACE in Figure 1.2. As illustrated, the goal of coreference resolution system is to construct

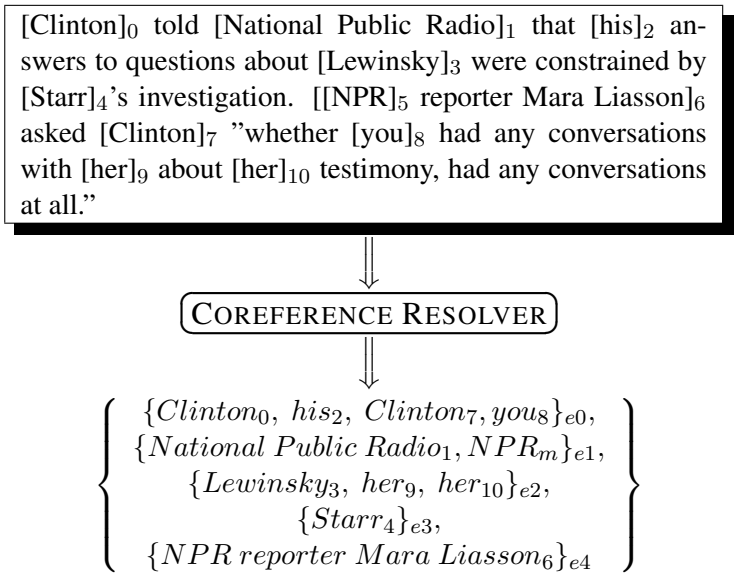

Figure 1.2: The task of coreference resolution

all the coreference links between between referential expressions. The reflexive, transitive closure over these links generates equivalence classes of expressions (or **coreference chains**), from which entities can be abstracted. A graphical representation of the coreference relation is given in Figure 1.3: all the possible coreferential links are represented with dashed lines, but only the solid lines describe the coreference relation for the example above.

[5] See (Bagga and Baldwin, 1998) for an example of work on cross-document coreference.

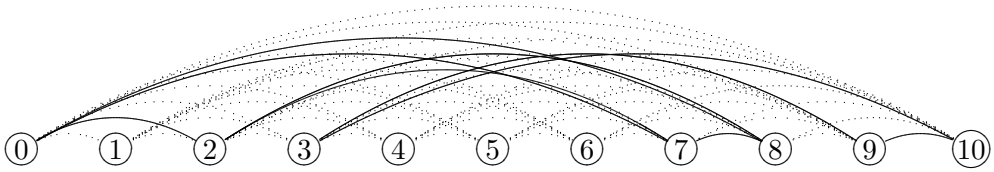

Figure 1.3: An example set of coreference relations

Clearly, the task of coreference is overall a much harder task than anaphora resolution. Intuitively, it is harder because we have to do more than linking a mention to (a previous mention of) its entity: we have in fact to predict the entities themselves. More technically, the complexity of the coreference resolution problem is *exponential* in the number of mentions: the search space is the set of all mutually disjoint subsets that can be created over the set of mentions. The problem of coreference resolution is indeed equivalent to the *set partitioning problem*, and its search space takes the form of a Bell Tree (Luo, 2005). Thus, the above example with only 10 mentions generates $115,975$ possible partitions (i.e., the 10^{th} Bell number). By contrast, the complexity of the anaphora resolution problem is merely *square* in the number of mentions: the candidate set for each anaphor is at most the set of mentions that precede it.

1.1.3 Relation between the two tasks

Searching the set of possible coreference partitions might be feasible for small documents, but it quickly becomes untractable for documents that have large numbers of mentions. One common, and rather intuitive way to deal with this problem is to reduce the task of coreference resolution to that of anaphora resolution.

There is indeed an obvious relation between between the two tasks. There is a sense in which anaphora resolution (especially, in the restricted definition used in NLP) is a strict sub-problem of coreference: it is coreference resolution restricted to a subset of expressions (namely, the pairs of expressions that are in an antecedent-anaphor relation). Inversely,

coreference resolution can be regarded as a sequence of anaphora resolutions, provided that we make the assumption that any mention in a chain but the head of the chain is considered "anaphoric". (Similarly, this makes any mention of a non-singleton chain an antecedent of the coreferential mentions that come after.) This is somewhat of a simplification, since some expressions like proper names (e.g., *Bill Clinton*) can appear further along in a chain without being strictly anaphoric to any previous mention. This assumption basically conflates the notion of being *anaphoric* with that of being *discourse-old* (e.g., Prince (1981)).[6]

1.2 General motivations

1.2.1 Theoretical challenges

As noted, the problem of anaphora resolution has been on the computational linguistics agenda since the early days. Despite the decades of work, the problem is still far from being being solved. In fact, this problem alone has often considered one of the hardest problems there is within AI. Given the relation between the two tasks, a similar case could easily be made for coreference resolution.

What makes reference resolution such an intrinsically difficult problem? Basically, the main problem is that **most referential expressions are ambiguous**, in the sense that many expressions could be coreferential in a particular context, but not in some other. This is due to the fact that the semantic content of many referential expressions is highly underspecified: the extreme case is pronouns, which are basically compatible with any expression provided some minimal grammatical agreement conditions are met.

An additional problem is in the dependence of anaphora and coreference resolution on a **multitude of knowledge sources**. Numerous factors have indeed been advanced by linguists to account for reference resolution (Mitkov, 2002a). These range from morphosyntax (e.g., gender, number, case) to syntax (e.g., grammatical relations and binding

[6] In the rest of this dissertation, these two terms will be used interchangeably.

principles) to lexical and compositional semantics (e.g., semantic typing, selectional restrictions) to discourse structure to world knowledge.

Given the reliance on so many knowledge sources, progress in reference resolution systems is very much contingent upon progress on advances on other tasks that come earlier in the commonly assumed NLP pipeline, such as syntactic parsing, semantic roles, discourse parsing, etc. The problem is that currently **many of these knowledge sources are hard to predict** in a robust and non-noisy way. The problem of predicting semantic representations for texts is at its beginnings, let alone that of predicting full discourse structures. Even the problem of syntactic parsing isn't a solved one.

Even if we had perfect representations for sentences and discourses, we would still be facing the problem that **none of the sources is completely reliable**. Thus, most of the constraints identified by linguists are "soft" (i.e., defeasible) constraints, rather than "hard" (i.e., undefeasible) ones. It is indeed very hard to find constraints that work all the time. For example, (Hirst, 1981) provides several examples of gender and number mismatches with pronouns —although number and gender agreement are taken by many to be a hard constraint for English pronoun resolution.

A final problem is that **different referential expressions follow different resolution strategies**. What this means is that different sources seem to be more or less important depending on the expression. A common example is recency. While pronouns, due to their lack of semantic content tend to be resolved to a nearby antecedent, other anaphoric expressions like definites or abbreviated proper names tend to show more long-distance resolutions. On the other hand, (sub-)string matching is obviously very important for linking proper names, but less so for other types of nominal expression.

1.2.2 Practical importance

An interesting challenge in itself, the proper identification of the entities that are referred to is also important from a purely language engineering perspective. Numerous NLP tasks

could in principle —when they haven't already done so— benefit from the availability of good reference resolution systems. We here focus on five specific tasks.[7]

Information Extraction. Generally, the goal of an information extraction system (IE) is to automatically induce structured information (e.g., in the form of templates expressed in a formal language) from machine-readable text. For instance, one might be interested in extracting certain relations (e.g., live_in, born_in) holding between entities. Knowing about coreference is crucial for IE: coreference can be used to merge different information regarding the same entity that might have been extracted at different places in the document. The importance of coreference for IE is reflected in the inclusion of the coreference task, along with Named Entity Recognition (NER), as part of IE competition-based conferences such as MUC-6 and MUC-7 or ACE more recently.

Question Answering. The goal of a Question Answering (QA) system is to answer a natural language question from a collection of documents (such as the web or a local database). It is therefore a specific subtype of Information Retrieval (IR). One way to use reference resolution for QA consists in resolving references before the indexation of documents, potentially allowing easier matching of the question.

Automatic Summarization. Automatic Summarization is the task of producing summaries based either on a single document or by grouping information from different documents. Several researchers have proposed to use coreference information to "guide" summarization (Azzam et al., 1999). For instance, large coreference chains can thought as important topics in a document (i.e., information that should appear in a summary).

Machine Translation. A Machine Translation (MT) system has the goal of automatically translating text (or speech) from one *source* language to one or several *target* languages.

[7] Some of the discussion in this section is based on (Mitkov, 2002b) and (Ng, 2002).

One way that anaphora/coreference resolution is relevant for MT is in the translation of languages that show morphological discrepancies (e.g., some languages have grammatical gender while others don't). A concrete example is the translation of pronouns from French into English: a pronoun like *elle* for instance should be translated by *she* when referring to a person, but by *it* when referring to an inanimate object.

Natural Language Generation. Natural Language Generation (NLG) is the problem of producing natural language from a formal representation such as a knowledge base or a logical form. An important challenge for NLG is that of producing *coherent* texts. Barzilay and Lapata (2005) show that using coreference chains help improve the coherence of texts.

1.3 Robust reference resolution

In the last section, we have described some of challenges inherent to reference resolution. In the face of these problems, much of the earlier work in anaphora resolution has concentrated their effort in attempting to represent and process domain and linguistic knowledge (Hobbs, 1978; Brennan et al., 1987; Carter, 1987; Rich and LuperFoy, 1988; Carbonell and Brown, 1988). Some of these approaches were either targeting one particular knowledge source (e.g., Hobbs (1978) exploits syntactic configurations to resolve anaphoric pronouns) or trying to explicitly model all the different information sources at play in resolution (e.g., the multi-strategy approach of (Carbonell and Brown, 1988)). In either case, the approach proceeds by manually writing explicit rules, therefore requiring a considerable amount of human input. Often, additional human intervention was also present for correcting the output of the different preprocessing modules.

The main problem of these approaches lies in their brittleness, which prevents their integration into larger NLP interfaces like IE or QA systems. This has led first, to the development of knowledge-poor approaches (Dagan and Itai, 1990; Lappin and Leass, 1994; Kennedy and Boguraev, 1996; Baldwin, 1997; Mitkov, 1998), which use clever heuris-

tics based solely on shallow processing, and more recently to the development of machine learning approaches (McCarthy and Lehnert, 1995; Morton, 2000; Soon et al., 2001; Ng and Cardie, 2002a). An advantage of the latter on the former is their robustness and the fact that they are theoretically more sound. The drive toward robust approaches was further motivated by the emergence of cheaper and more reliable NLP tools (e.g., part-of-speech taggers and shallow parsers) and the availability of corpora annotated with coreference information and resources like lexical databases (e.g., Wordnet). With these resources also came better evaluation practices: most earlier systems were either not evaluated at all, or not evaluated against a common benchmark, making any comparison difficult.

With a few exceptions, most machine learning approaches to reference resolution have been *supervised* approaches.[8] Common to most of these approaches is that they recast both the tasks of anaphora resolution and coreference resolution as a very simple learning problem: that is, a **binary classification** problem. Specifically, annotated data are converted into pairs of potential anaphors and potential antecedents. These data instances are realized as feature vectors and are labelled with a target concept, e.g. values 1 or 0, indicating whether the mentions are coreferential or not. Learning consists in finding a set of weights (or model) which best determines the importance of each feature in predicting the correct labelling of the mention pairs. Once trained, the classifier is applied to label the mentions pairs that make up the test data. In general, this classification step is complemented by a search or **clustering algorithm**, whose role is to select a unique antecedent for anaphora resolution, or to merge the different coreferential links for coreference resolution. For anaphora resolution, this step is justified by the fact that we want to select the *best* antecedent. For coreference resolution, this is justified by the fact that we want filter out potentially conflicting links (i.e., links that violate transitivity). Two such algorithms have been commonly used: *closest-first* and *best-first*. Both of these select a unique antecedent for each "anaphor": the former picks the closest coreferential mention, while

[8] Unsupervised approaches include (Cardie and Wagstaff, 1999) and (Bean and Riloff, 2004).

the latter picks the antecedent that is associated with the largest probability score. These link-selection algorithms are often used in tandem with a particular sampling method of the training instances.

In addition to robustness, these rather simple machine learning techniques have resulted in significant gains in performance over hand-crafted systems.[9] These improvements are likely to come from different advantages offered by machine learning techniques. By definition, these techniques have built into them the ability to handle soft constraints —in the form of *features* which receive particular weights through training. Furthermore, some of these models (i.e., *discriminative* models) are particularly well-suited to problems that involve many, potentially conflicting knowledge sources.

1.4 Research objectives and contributions

Despite these improvements, these systems are still far from being perfect, and performance tends to plateau at accuracy scores in the 70% range for pronoun resolution, and f-scores in the 60% range for coreference resolution. This brings us to the main objective of this dissertation, namely to improve on these existing state-of-the-art systems. In the following, we identify several factors that are potential limitations of the existing systems, and suggest a set of alternatives that will be pursued in the rest of the dissertation:

Model type: As noted, most approaches use binary classification, in which each pair of mentions is classified as coreferential or not. But on closer inspection, it seems that classification is not so well-suited to the problem of anaphora/coreference resolution. Focusing for now on the problem of anaphora resolution, note that the ultimate goal of this task is to find the "best" antecedent among a set of candidates, and not crucially to find all the "good" antecedents. That is, we are ultimately interested in

[9] This has been shown for coreference resolution in the context the MUC-6 and MUC-7 competitions. See also (Preiss, 2002) and (Kehler et al., 2004a) for some comparisons between hand-crafted and machine learning anaphora resolution systems.

learning a *ranking* function over the set of antecedent candidates rather than a binary classification function. The crucial appeal of a ranking approach is that it brings the comparison between antecedent candidates inside the training criterion, rather than deriving it from the classifier's probabilities. This results in better antecedent selection capabilities. Assuming that one recasts coreference resolution as a sequence of anaphora resolutions, using a ranker has also the potential to improve on this larger task.

Single model: Most of the existing approaches to anaphora resolution and coreference resolution proceed by learning a single classification model, therefore giving a uniform treatment to different types of anaphors. This is problematic, given that different anaphoric expressions are sensitive to different factors in different ways. What we propose instead is to build several models for different anaphoric expressions: these models use different feature sets and different sampling strategies during training. This "distributed" approach is not entirely new, and specialized models have been proposed both in the context of coreference resolution (Morton, 2000) and pronoun resolution (Ng, 2005a). The originality of our contribution here is that we propose separate ranking models.

Decision locality: The two previous problems concern the type of model that was learned. A further potential weakness is in the application of the model during testing. This problem is particularly critical to coreference resolution. For that task, the classifier model is traditionally used in combination with a *separate* clustering algorithm that is responsible for coordinating pairwise decisions into coreference chains. The problem with this approach is that the clustering decisions are made independently of one another, which means that only *local* coherence is ensured. Ideally, one would also like to enforce a more *global* notion of coherence. The decision of merging a mention into a chain should depend on how well it matches the entity as a whole (McCallum and Wellner, 2003). A related problem with this approach is that classification and

clustering are optimized separately, which means that improvements brought to the classification model might not lead to overall performance gains (Ng, 2005b).

There are various ways to address this issue. One can actually learn a different type of model where coreference decisions are directly conditioned on entities (i.e., chains), rather than on mentions (Morton, 2000; Luo et al., 2004; Culotta et al., 2007). This has the advantage of allowing one to define larger features, and therefore ensuring better global coherence. But this also makes the search process much more complicated. One alternative to these *global models* is to still train *local models* (i.e., mention-based models), but to incorporate *global constraints* during inference. One type of global constraint that is likely to be useful for coreference resolution are transitivity constraints: these constraints can be used to ensure that the consistency between pairwise coreference assignments. This global inference can be cast as an optimization problem, in particular as an integer linear program (ILP), and can be solved using standard optimization tools. This general framework has been developed in the work of Dan Roth (e.g., Roth and Yih (2004)).

Knowledge prediction and integration: Reference resolution depends on many different information sources. Yet most systems have to date only been relying on very small sets of shallow features: for instance, Soon et al. (Soon et al., 2001) use 12 features. Consequently, a prevailing view is that improving anaphora/coreference resolution requires the incorporation of more sophisticated knowledge sources into the models (in particular, syntactic and semantic ones). Unfortunately, attempts at adding in more information sources have been overall disappointing, leading to small improvements (Ponzetto and Strube, 2006; Yang et al., 2006; Ng, 2007), but also to degradation (Kehler et al., 2004a; Ng and Cardie, 2002b; Denis and Kuhn, 2006) in performance.

Predicting linguistically rich information from raw text is indeed challenging, as noted earlier. Given that the extracted information is likely to be noisy, the issue of how to best incorporate this information becomes crucial. There are two typical

ways of integrating knowledge sources into a system: (i) a pre- or post-processing module (this corresponds to the traditional pipeline architecture), (ii) in the form of features. None of these views is perfect. The first approach faces the problem of error propagation: error made by the upstream model tend to propagate into the downstream model. Integrating information as features alleviates error propagation somewhat, since the noise is already present in the training. But this approach might face the problem of feature "washout", where some normally "good" features do not have their discriminative power due to the presence of many other features. At a more abstract level, the problem is that there are often complex, mutual dependencies between the outcomes of the upstream and downstream models. Failing to encode these dependencies means that the upstream model is going to over-constrain the downstream model.

One way to handle these dependencies is to cast the two problems as a *joint* problem. In this dissertation, we focus on predicting two types of information likely to improve coreference resolution: (i) discourse status information (aka anaphoricity), and (ii) named entity type. Intuitively, we only should identify antecedents for the mentions which are likely to have one (Ng and Cardie, 2002b), and we should only make a set of mentions coreferent if they all have the *same* entity type (eg, PERSON or LOCATION). Specifically, we show that the linear programming framework allows these models to be optimally integrated, through the use of mutual consistency constraints, with a coreference model.

1.5 Dissertation outline

Chapter 2 This first chapter provides the background for the rest of the dissertation. We start by presenting a brief history of the various trends of research that have dominated the field of reference resolution, focusing in particular on the recent shift to machine learning approaches. This chapter also discusses the various corpora and

evaluation metrics development for anaphora and coreference resolution.

Chapter 3 This chapter presents a new approach to the problem of anaphora resolution. In particular, we propose a *ranking* approach to pronoun resolution as an alternative to the traditional classification-based approach. We start by motivating the ranking approach in the context of maximum entropy models: in particular, we show that the ranker offers a potentially better way to model the task by directly including the comparison between antecedent candidates as part of its training criterion. In order to test this hypothesis, we run various experiments comparing the ranker against various baseline classification-based systems: in particular, the standard binary classifier discussed above, and the related twin-candidate approach of (Yang et al., 2003; Yang, 2005). These experiments reveal that the ranker provides significant performance improvements over the other systems and is also computationally more attractive in terms of training time and learning rate.

Chapter 4 This chapter extends the ranking approach to the larger problem of coreference resolution. Roughly, the goal is to see whether the better antecedent selection capabilities offered by the ranking approach can also benefit in the larger coreference resolution task. The extension is two-fold. First, we design various specialized ranker models for different types referential expressions (e.g., pronouns, definite descriptions, proper names). Besides its linguistic appeal, this division of labor has also the potential of learning better model parameters. Second, we augment these rankers with a model that determines the discourse status of mentions and that is used to filter the "non-anaphoric" mentions. As shown by various experiments, this combined strategy results in significant performance improvements over the standard approach on the coreference task.

Chapter 5 In this chapter, we show how the task of coreference resolution can be recast as a linear optimization problem. In particular, we use the framework of Integer Linear

Programming (ILP) to: (i) combine the predictions of three local models (namely, a standard pairwise coreference classifier, a discourse status classifier, and a named entity classifier) in a joint, global inference, and (ii) integrate various other global constraints (such as transitivity constraints) to better capture the dependencies between coreference decisions. Tested on the ACE datasets, our ILP formulations deliver significant f-score improvements over both a standard pairwise model, and various models that employ the discourse status and a named entity classifiers in a cascade. Improvements were found across the three different evaluation metrics: MUC, B^3, and CEAF.

Chapter 2

State of the art: approaches and evaluation

This chapter describes the state of the art in computational approaches to reference resolution and serves as the background for the rest of this dissertation. First, we present in Section 2.1 a generic algorithm that describes most of existing systems proposed for anaphora resolution and coreference resolution. In Section 2.2, we then give a brief historical overview of the different approaches offered to the problem, focusing on some milestone approaches that reflect the evolution of the field. Over the last two decades, research on reference resolution has seen the emergence of machine learning methods. The next section, Section 2.3, describes the "standard" learning approach to reference resolution: most of learning approaches have in common that they recast the problem as a binary classification problem. Some variations on this binary classification scheme are also discussed, along with the inherent limitations of this approach. As discussed in Section 2.4, the emergence of machine learning approaches to reference resolution has seen the increasing availability of corpora annotated with coreference information and the development of precise evaluation metrics: the most commonly used corpora and metrics are described.

2.1 A generic algorithm

Algorithm 1 RESOLVE

Input: A document D
Output: A set of coreference links L_D for D

```
// 1. Identification of mentions in D
```
$\mathcal{M} \Leftarrow \{m \mid m$ is a referential mention in $D\}$
$\mathcal{A} \Leftarrow \mathcal{M}$

```
// 2. Characterization of mentions
```
for all m_i in $\mathcal{M}$ **do**
  Compute a set of values for $\{k_{i_1}, k_{i_2}, \ldots, k_{i_n}\}$ from n knowledge sources
end for

```
// 3. Anaphoricity determination (Optional)
```
$\mathcal{A} \Leftarrow \mathcal{A} \setminus \{m \in \mathcal{A} \mid m$ is not anaphoric$\}$

for all m_j in $\mathcal{A}$ **do**
```
  // 4. Generation of antecedent candidates
```
  $\mathcal{C}_j \Leftarrow \{m \in \mathcal{M} \mid m$ lies in the scope of $m_j\}$

```
  // 5. Filtering (Optional)
```
  $\mathcal{C}_j \Leftarrow \mathcal{C}_j \setminus \{m_i \in \mathcal{C}_j \mid m_i$ violates a coreference constraint with $m_j\}$

```
  // 6. Scoring/Ranking
```
  Score or rank each m_i in $\mathcal{C}_j$ and sort $\mathcal{C}_j$ w.r.t. the score

```
  // 7. Search/Clustering
```
  Select an antecedent for m_j from $\mathcal{C}_j$

end for

Despite important differences, the vast majority of the previous reference resolution systems (be they hand-crafted or corpus-based) can seen as instantiations of a generic algorithm given in Algorithm 1. The RESOLVE algorithm and its description are adapted from (Ng, 2002). This algorithm takes a document D in raw text format as input, and computes a set of anaphoric/coreferential links L_D for D. These links L_D can be encoded in the form of pairs of mentions: in the case of coreference resolution, the chains are obtained through

simple reflexive, transitive closure over these pairs.[1]

As noted in Chapter 1, a common approach to the problem of coreference resolution is to reduce it to the simpler problem of anaphora resolution. Under this view, each pair of mentions that is coreferential is in effect an $\langle antecedent, anaphor \rangle$ pair. Present (at least implicitly) in most work on coreference resolution, this assumption is directly embodied in the RESOLVE algorithm, since the algorithm initially treats each mention in the text as a possible anaphor (step 1) and tries to find its antecedent(s) (steps 4-7).

Let us now look at the different steps of RESOLVE in more detail. The first step consists in the **identification of the referential mentions** in the document D: $\mathcal{M}$ is the list of all the mentions in D. Concretely, this means finding the different nominal and pronominal expressions in D that have referential content. This leaves out pleonastic uses of pronouns, for instance. This step is performed automatically via a preproccessing module (built around a NP chunker or a full parser, and a named entity recognizer) or by selecting the mentions whose boundaries are encoded in a pre-existing corpus. Most published research are actually unclear regarding this point, and even when they perform automatic mention detection, rare are the authors that report the scores for this component (although the performance of this preprocessing module can have drastic effects on the final performance of the resolution system).[2] Note that another operation takes place during this first step: the set of anaphoric expressions $\mathcal{A}$ is initialized to $\mathcal{M}$; that is, every mention is at first assumed to be anaphoric.

The second step is the **characterization of mentions**, and involves determining and extracting the different knowledge sources that characterize a mention and that might be relevant to its linking to the other mentions in the document. Existing systems differ along two dimensions with respect to this step. Again, some approaches are fully automatic relying on some preprocessing modules (e.g., part-of-speech tagging, named entity recognizer, parsing, etc.), while others use gold standard information if using a corpus containing this

[1] This is the format used on the MUC corpora. See our discussion in Section 2.4.

[2] In this dissertation, we will always assume "true" mention boundaries as given by an annotated corpus.

information or manually correct the outputs of the preprocessing modules. They also differ in the level of sophistication of the extracted information, ranging from knowledge-rich to knowledge-poor (this will be discussed in more detail in Section 2.2).

The third step is that of **anaphoricity determination**, and it consists in the filtering from $\mathcal{A}$ of the mentions that are not anaphoric. These mentions, by definition, do not have an antecedent and hence shouldn't be resolved. This step is always (by definition) performed as part of anaphora resolution, but it is not always performed for coreference resolution, in which case all mentions in effect remain possible anaphors.

While the previous steps can be seen as preprocessing steps and are performed at the level of the entire document, the last four steps operate the resolution phase and operate on and are performed at the level of each mention m_j in $\mathcal{A}$. The fourth step realizes the **generation of antecedent candidates** for each possible anaphoric mention. By default, the set of candidates $\mathcal{C}_j$ are the mentions that linearly precede m_j in the document. For pronoun resolution, this set is sometimes restricted to the set of mentions that lie within a certain window of sentences (typically, the current and two or three preceding sentences).

The next step is another **filtering** step, and amounts to reducing the space of antecedent candidates for the given anaphor. This step involves removing from $\mathcal{C}_j$ some unlikely antecedent candidates based on a set of hard constraints. This step is often used in pronoun resolution systems, where some constraints such as gender and number agreement or binding principles are taken to be inviolable, hard constraints. This step is however not present in every anaphora resolution or coreference resolution system.

The goal of the **scoring/ranking** step is basically to order the candidates that made it through the previous steps. The ordering is obtained based on a set of rules or soft constraints and takes one of the following forms. In some approaches, each candidate m_i receives an individual numerical score (e.g., a probability for statistical approaches) that reflects the likelihood of m_i and m_j to be in a coreferential or in an anaphoric relation. The scores obtained for each candidate can then be used for sorting them. In other approaches,

the candidates are directly ordered through the application of various preference rules or discourse principles.

The last step of the algorithm is **searching/clustering**: this step results in the actual selection of an antecedent from the ranking obtained from the previous step. In principle, the list of candidates can be empty, in which case m_j is left unresolved, meaning that m_j is in fact not anaphoric. If it isn't empty, the list is searched. The search proceeds either by picking the candidate with the highest score or by going through the list in some specific order (e.g., reverse linear order) and picking the first candidate that meets a threshold score. Some coreference resolution systems also allow for the selection of several antecedents for m_j. Through transitive closure, these various antecedent selection techniques implicitly amount to imposing a partition over $\mathcal{M}$.

2.2 Brief history

Historically, research in computational reference resolution has seen the succession of three main paradigms. With respect to the RESOLVE algorithm, these different approaches differ along two three main dimensions: (i) the level of automation in the preprocessing steps 1-3, (ii) the amount of sophistication in the sources present in step 2, and (iii) the type of methods used for resolution steps 5 and 6.[3]

2.2.1 Knowledge-based systems

The first reference resolution systems, developed from the sixties through the eighties, were hand-crafted **knowledge-based systems**. Viewed as a whole, this body of work can be described as an attempt to algorithmically model the linguistic knowledge (and sometimes also domain and world knowledge) influencing anaphora resolution. These different knowledge sources are encoded in the form of rules that are manually engineered according to lin-

[3]The review proposed in this section is by no means exhaustive. The interested reader is referred to (Hirst, 1981), (Mitkov, 2002a) and (Ng, 2002) for more complete panorama.

guistic theories. Most of these approaches rely on a distinction between (hard) *constraints* and *preferences* during the resolution steps 5-6. Also typical of these approaches is that they assume perfect input: texts are pre-analyzed or at least corrected by human experts. Not all these systems were actually implemented, and when they were, they are usually manually tested.

Within knowledge-based approaches, one can further distinguish between approaches that focus on the detailed modelling of *one* type of linguistic knowledge (either syntax, semantics, or discourse), and approaches that try to combine the effects of multiple knowledge sources. An example of the former is the **syntax-based** approach of (Hobbs, 1976, 1978). In these two papers, Hobbs proposes a "naive" algorithm for pronoun resolution that solely relies on syntactic and morphological information. This algorithm assumes full syntactic trees for the input text and uses (i) a morphological filter to rule out unlikely antecedents and (ii) a tree traversal search to find the "best" antecedent. This algorithm is interesting both for its extreme simplicity: it relies solely on binding and agreement constraints, and its good performance (especially for intra-sentential resolutions): (Hobbs, 1978) reports accuracy scores as high as 88% based on manual evaluation. The main drawback of this approach is that it requires full syntactic parses, which are still hard to produce in a robust and non-noisy way.

Another type of knowledge-based approach can be found in the work of (Brennan et al., 1987) and (Grosz et al., 1995). Often called **discourse-based**, these approaches are based on Centering Theory, a theory of discourse inspired by early works of Barbara Grosz and Candice Sidner (e.g., (Grosz and Sidner, 1986)). Roughly, centering proposes a set of constraints and principles whose aim is to track down the focus of attention of discourse participants. The antecedent candidate in focus is the most salient to be referred to by the current pronominal anaphor or definite description. Interestingly, salience in this theory ranking is primarily determined on the basis of surface syntactic information. Overall, these "attention-based" approaches have had rather mixed results: for instance, (Walker, 1989)

shows that the algorithm of (Brennan et al., 1987) underperforms against Hobbs' naive approach. See (Kehler, 1997) and (Beaver, 2004) for some interesting criticisms of Centering, and (Tetreault, 2001) for a corpus-based evaluation of these centering-based algorithms. Finally, note that other discourse-based approaches have been proposed to handle anaphora resolution (Hobbs, 1979; Kehler, 2002; Asher and Lascarides, 2003). Instead of viewing anaphora resolution as independent process (as is done in Centering), these "coherence-driven" approaches instead view it as a by-product of the establishment of discourse relations. Several systems can be seen as attempts to implement some of these ideas (e.g., Cristea et al. (1998)).

While the two previous approaches focus on modelling a particular type of information source, (Carbonell and Brown, 1988) instead take the view that anaphora resolution can be best accomplished through the combination of a set of strategies. Thus, this **multi-factor** approach relies on a set of constraints and preferences, which are syntactic, semantic, and pragmatic, to determine the antecedent of an anaphor. For instance, the syntactic constraints include gender and number agreement, while the syntactic preferences include topicalization and grammatical parallelism. The semantic constraints include selectional restrictions, while the semantic preferences include thematic role parallelism. A characteristic of this approach is that conflicts between various applicable preferences are not resolved: in those cases, the anaphor is considered to be truly ambiguous.

2.2.2 Heuristics-based systems

Difficult and expensive to build, the knowledge-based systems suffer from the additional problem of their lack of robustness which makes them difficult to port to other domains and languages or to incorporate into larger NLP interfaces. These limitations, combined with the development of cheaper and more reliable NLP tools (such as part-of-speech taggers and chunkers), led many researchers to investigate alternative solutions. The nineties thus saw the emergence of **heuristics-based systems**, also known as knowledge-poor systems.

These systems adopt a more engineering approach and can be seen as an attempt to make the most of limited and possibly noisy information by using carefully designed heuristics. Less concerned by theoretical motivations, these systems achieve performance that are comparable to their knowledge-based counterparts, while gaining in design simplicity and robustness.

An example of this type of approach is the Resolution of Anaphora Procedure (RAP) proposed by Lappin and Leass (1994) which is a salience-based algorithm for resolving third person pronouns. The algorithm works by computing a salience measure for each antecedent candidate based on several salience factors determined in terms of grammatical role, parallelism of grammatical roles, frequency of mention, and sentence recency. Each of these factors is associated with an initial, pre-defined weight (the salience sentence recency is assigned the initial weight of 100, subject emphasis 80, etc.). These different weights get degraded as sentences in the discourse get processed. The salience of each candidate is computed as the sum of the salience values of the elements in its current chain. Eventually, the candidate with the highest salience measure is chosen as the antecedent. Using the perfect output from a morphological analyzer and a full syntactic parser, Lappin and Leass (1994) report an accuracy score 86% (on 360 pronoun occurrences).

Kennedy and Boguraev (1996) propose an interesting extension to Lappin and Leass' approach. By contract to RAP, their system does not require in-depth and full syntactic parsing, but relies only on POS tagging and grammatical functions of lexical items. They reported 75% success, on a random selection of documents from different genres (from press releases to web pages). Another extension of RAP was made by Mitkov (1998), who investigates a wider list of different salience factors ("indicators" in Mitkov's terms). The weighting scheme is this approach is different in that candidates are assigned a score (2, 1, 0, -1) for each salience indicator. Mitkov's indicators are related to salience (e.g., definiteness, indefiniteness, giveness, repetition), structural matches (e.g., collocation, sequential structure), distance. This approach was evaluated on a small corpus of technical manuals

(containing only 223 pronouns), where it achieves a success rate of 89.7%.

2.2.3 Machine learning systems

Although both knowledge-based and heuristics-based approaches have still been pursued, the last decade of research in reference resolution has seen the emergence of statistical and **machine learning systems**. These new approaches typically also use limited knowledge sources (that is, they are also "knowledge-poor"), but they do away with manually engineered rules or heuristics relying instead on numerical methods to determine the importance of these sources in the resolution decisions. This makes these systems easier to design and to port to other domains and languages, and theoretically more appealing since they are grounded in a mathematically sound framework. This shift, also present in other areas of NLP, has been made possible due to increasing availability of corpora annotated with anaphoric/coreference information. Some of these have been developed in the context of various IE shared-task competitions such as the Message Understanding Conference (MUC) and the Automatic Content Extraction (ACE). These competitions have somewhat redefined the research agenda by putting the emphasis on the larger task of coreference resolution (arguably more useful to IE) and have led to the development of better evaluation standards.

Supervised approaches

Unlike manual approaches, machine learning approaches to coreference resolution induce a model that determines the probability that two NPs are coreferent from annotated data automatically via the use of *learning algorithms*. They can be characterized in terms of the knowledge sources being employed (represented as *features*), the method of training data creation (or *sampling*), as well as the the *clustering algorithm* being chosen.[4]

[4] At the core of supervised learning is the so-called *inductive learning hypothesis* (Mitchell, 1997), p.23, according to which:

> Any hypothesis found to approximate the target function well over a sufficiently large set of

Since the work in this dissertation directly builds upon previous supervised machine learning approaches, we present these approaches in more detail in Section 2.3.

Unsupervised approaches

While most of the most of the work in learning-based reference resolution has used supervised learning techniques, there has also been at least two attempts at developing unsupervised approaches. By definition, these approaches do not make use of annotated data for training their systems.

The first approach is proposed in Cardie and Wagstaff (1999). These authors explicitly view coreference as a clustering task and use a right-to-left single-link clustering algorithm to partition the set of mentions into coreference equivalence classes. The clustering algorithm uses a distance metric between two mentions that is a linear combination of the incompatibility scores computed for the two mentions. Merging is considered whenever the distance is less than the predefined clustering radius. The knowledge sources used in Cardie and Wagstaff (1999) include: lexical (e.g. head noun match, word overlap), syntactic (e.g. gender, number, animacy, apposition), semantic (e.g. WordNet class compatibilities), and positional (e.g. number of intervening noun phrases between the two NPs under consideration) features. Like in the heuristic-based approaches, the weight associated with each feature is manually determined. This approach was evaluated on MUC-6 data set, and obtained 48.8% recall and 57.4% precision.

Another unsupervised approach is proposed by Bean and Riloff (2004) in their BABAR system. The focus of this study is on the incorporation of contextual (or thematic) role knowledge to identify the coreferential pairs. BABAR employs IE techniques to represent and learn role relations, and uses unsupervised learning to acquire this knowledge from plain texts. Learning starts by generating a set of "seeds", which are cases of anaphor-

training examples will also approximate the target function well over other unobserved examples.

antecedent pairs that can be easily be resolved (e.g., by string matching). BABAR then applies the AutoSlog system of Riloff (1996) to the unannotated training texts to generate a large set of case frames coupled with a list of extracted noun phrases. For coreference resolution, BABAR utilizes three contextual role sources derived from the caseframe data: (i) the caseframe network (i.e., the caseframes that co-occur in anaphor-antecedent pairs), (ii) lexical caseframe expectations (i.e., two coreferential mentions are substitutable for each other in their caseframes), and (iii) semantic caseframe expectations (i.e., the same as (ii) but based on the semantic classes of the mentions). The resolution uses seven additional sources including gender/number/semantic matching, distance, recency, etc. The system was tested on the definite and pronominal anaphors of MUC-6 corpus. The main positive result of this study is that the unsupervised-learned contextual roles are able to improve recall of 15% in the resolution of pronominal anaphors. [5]

2.3 Standard machine learning approach

2.3.1 Reference resolution as binary classification

The standard approach recasts the task of reference resolution as a binary classification problem in which pairs of mentions are labelled as either coreferential or not. For coreference resolution, this classification phase is combined with a clustering alogrithm that is responsible for merging the links identified into coreference chains. As a representative example of this approach, we describe (Soon et al., 2001): this approach is typically used as a baseline for comparing other approaches. Recent variations on and departures from this original approach are also discussed in the next sections.

[5] See also Haghighi and Klein (2007) for a new and very promising unsupervised approach to coreference.

Model

This approach tackles coreference in two steps by: (i) estimating the probability, $P(\text{COREF}|\langle\pi, \alpha_i\rangle)$, of having a coreferential outcome given a pair of mentions $\langle\pi, \alpha_i\rangle$, and (ii) applying a selection algorithm that will single out a unique candidate out of the subset of candidates α_k for which the probability $P(\text{COREF}|\langle\pi, \alpha_k\rangle)$ reaches a particular value (typically .5). For building their classifier, (Soon et al., 2001) use the C4.5 tree induction system as their learning algorithm.

Training

Training instances are constructed based on pairs of mentions of the form $\langle\pi, \alpha_i\rangle$, where π and α_i are the descriptions for an anaphoric mention and one of its candidate antecedents, respectively. Each such pair is assigned either a label COREF (i.e. a positive instance) or a label NOT-COREF (i.e. a negative instance) depending on whether or not the two mentions corefer. In generating the training data, we create for each anaphoric mention: (i) a *positive instance* for the pair $\langle\pi, \alpha_i\rangle$ where α_i is the closest antecedent for π, and (ii) a *negative instance* for each pair $\langle\pi, \alpha_j\rangle$ where α_j intervenes between α_i and π.

Feature set

The system proposed by Soon et al. (2001) use a set of 12 simple features, describing: (i) the anaphoric mention π, (ii) the antecedent candidate α, and (iii) the relation between the two mentions. These features are informally described in Table 2.1.

Resolution

Once trained, the classifier is used to select a unique antecedent for each anaphoric pronoun in the test documents. In the Soon et al. (Soon et al., 2001) system, this is done for each pronoun π by scanning the text right to left, and pairing π with each preceding mention α_i. Each test instance $\langle\pi, \alpha_i\rangle$ thus formed is then evaluated by the classifier, which returns a

	Feature	Description
1.	ana_pro	π is a pronoun $\{1,0\}$
2.	ana_def_np	π is a definite NP $\{1,0\}$
3.	ana_dem_np	π is a demonstrative NP $\{1,0\}$
4.	ante_pro	α is a pronoun $\{1,0\}$
5.	distance	Distance between π and α in sentences $\{0,1,2,\dots\}$
6.	str_match	π and α have identical string $\{0,1\}$
7.	both_pn	π and α are both proper names $\{0,1\}$
8.	gender_agr	π and α have the same gender $\{0,1\}$
9.	number_agr	π and α have the same gender $\{0,1\}$
10.	sem_class_agr	π and α have the same semantic class $\{0,1\}$
11.	alias	π is an alias (e.g., acronym) of α $\{0,1\}$
12.	appositive	π is an appositive of α $\{0,1\}$

Table 2.1: Feature set used by Soon et al. (2001)

probability representing the likelihood that these two mentions are coreferential. Soon et al. (Soon et al., 2001) use "Closest-First" selection: that is, the process terminates as soon as an antecedent (i.e., a test instance with probability $> .5$) is found or the beginning of the text is reached.

2.3.2 Variations on this approach

Learning algorithm

A lot of implementations of the pairwise classifier have used Decision Trees (McCarthy and Lehnert, 1995; Soon et al., 2001; Ng and Cardie, 2002a), but other types of learning algorithms have also been used more recently. For instance, (Morton, 2000; Kehler et al., 2004b) use maximum entropy models, while (Ponzetto and Strube, 2006) use SVMs. See (Uryupina, 2004) for a comparison of different learning algorithms.

Feature set

As noted above, Soon et al. (2001) only use 12 features. A number of recent works have focused on enhancing the feature set used by pairwise coreference classifier. For instance,

Ng and Cardie (2002a) expand the feature set to a deeper set of 53: these features allow more complex NP string matching operations, as well as finer-grained syntactic and semantic compatibility tests. More recently, several approaches have tried to include finer-grained syntactic information (Denis and Kuhn, 2006; Yang et al., 2006), others to use richer semantic features (e.g., selectional restrictions, semantic roles) (Kehler et al., 2004a; Ponzetto and Strube, 2006; Ng, 2007).

Sampling method

Various alternatives have been proposed to the sampling method proposed in Soon et al. (2001). Most of these methods primarily differ in the creation of the positive (i.e., coreferential) instances. In McCarthy and Lehnert (1995), positive instances are created for each anaphoric mention paired with *each* of its antecedents, while negative instances are created by pairing each mention with each of its preceding non-coreferent noun phrases. This results in many more instances being created, and can potentially make the training process inefficient. The approach of Soon et al. (2001) is actually posterior to McCarthy and Lehnert (1995), and was presented as an attempt at reducing training times. Another method is proposed in Ng and Cardie (2002a). This method generates positive instances for each anaphoric mention and its most *confident* antecedent, which is defined as: (i) the closest preceding antecedent if the anaphor is a pronoun, but (ii) the closest *non-pronominal* antecedent if the anaphor is a non-pronominal anaphor. Negative instances are generated as Soon et al. (2001). Uryupina (2004) further refines the method used by Ng and Cardie (2002a) by providing different samplings for different NP types (e.g., proper names, definite descriptions).

Link selection

A number of different link-selection approaches have also been proposed; these often work in tandem with a specific sampling method during training. Ng and Cardie (2002a) proposes

a "Best-First" clustering algorithm as an alternative to the "Closest-First" algortihm of Soon et al. (2001). The "Best-First" algorithm selects as the antecedent the (closest) mention that has received the highest coreference probability from its set of preceding coreferent mentions. McCarthy and Lehnert (1995) uses an "Aggressive-Merge" clustering, in which each mention is merged with all of its preceding coreferent mentions. Note that the later is likely to yield higher recall, while the two previous algorithms are likely to be better in precision.

2.3.3 Limitations and recent developments

Model type

A potential drawback of pairwise classification is that it treats each antecedent candidate as a separate, independent event, and as such fails to capture the dependencies between the different candidates (Yang et al. (2003)). Ideally, one would like to make the competition between these different candidates part of training and directly learn a ranking function over this candidate set. Such a ranking approach is explored for anaphora resolution in Chapter 3 and extended for coreference resolution in Chapter 4.

Single model

Within the standard approach, anaphora resolution and coreference resolution proceed by learning a single, monolithic classification model. This in effect amounts to giving a uniform treatment to different types of anaphors.[6] This is problematic, given that different anaphoric expressions are sensitive to different factors in different ways. In Chapter 4, we propose to build several ranking models for different anaphoric expressions: these models use different feature sets and different sampling strategies during training.

[6]Two noticeable exceptions are (Morton, 2000) and (Ng, 2005a), who propose using separate (classification) models for different types of expressions.

Decision locality

Another weakness of the standard approach is that it fails to account for dependencies between coreference decisions. This is true both during the training of the model (since the model is trained solely based on pairs of mentions), and during its application (since the clustering decisions are made independently). For instance, note that with the clusterings described above, nothing prevents a situation like the following (where "$=_c$" stands for "corefer"):

(2.1) $A =_c B, B =_c C, A \neq_c C$

This limitation affects both anaphora resolution and coreference resolution, but it is especially important for coreference resolution, where one would like to ensure that the set of mentions in an entity forms a coherent whole. Part of the problem of the standard approach is that the classification and the clustering steps are optimized separately. This means in turn that any improvement brought to the classifier are not guaranteed to produce overall improvements (Ng, 2005b).

Different "global" approaches have been proposed to tackle these problems. Some approaches have tried to alleviate these problems while still relying on pairwise classifications of mentions. An earlier attempt is provided by (Morton, 2000) and relies on using a discourse model. (Kehler, 1997) uses Dempster's Rule to combine pairwise coreference probabilities to compute the score of the global partition. A more sophisticated approach is proposed by (Luo et al., 2004) and (Luo, 2007): these authors model the coreference problem using a Bell tree and use beam search for constructing the final tree during testing. (Ng, 2005b) proposes an approach where the outputs of different resolvers are reranked. Other approaches propose to directly learn a global model where coreference decisions are directly conditioned on entities (i.e., chains), rather than on mentions (Morton, 2000; Luo et al., 2004; Culotta et al., 2007). A different type of global approach is proposed in Chapter 5.

Knowledge prediction and integration

Reference resolution depends on many different information sources. Yet most existing approaches have only been using very limited information sources. Furthermore, many recent attempts at incorporating more information sources have been disappointing, leading to degradation in performance (Kehler et al., 2004a; Ng and Cardie, 2002b; Denis and Kuhn, 2006). The main problem faced by these approaches is the extraction of linguistically rich information from raw text is very challenging, hence error-prone. This raises the following question: how to best incorporate this potentially imperfect information? For the most part, previous approaches have incorporated additional information sources either as features or as a pre- or post-processing module. Both of these approaches are however problematic. The first approach faces the problem of error propagation: error made by the upstream model tend to propagate into the downstream model. Integrating information as features alleviates error propagation somewhat, since the noise is already present in the training. But this approach might face the problem of feature "washout", where some normally "good" features do not have their discriminative power due to the presence of many other features. At a more abstract level, the problem is that there are often complex, mutual dependencies between the outcomes of the upstream and downstream models. Failing to encode these dependencies means that the upstream model is going to over-constrain the downstream model. Chapter 5 proposes a more elaborate way to combine different models for coreference.

2.4 Corpora and evaluation

The recent improvements in robust reference resolution have been made possible due to the increasing availability of large annotated corpora with anaphoric/coreference information and the related development of rigorous numerical evaluation standards. This section introduces the main datasets available and describes the most commonly used evaluation metrics

for anaphora and coreference resolution.

2.4.1 Corpora

The largest and most commonly used corpora for developing and evaluating anaphora/coreference systems are the MUC-6 corpus (muc, 1995), MUC-7 corpus (muc, 1998), and the ACE corpora.[7] These corpora have been created in the context of two IE government-funded competitions: respectively, the Message Understanding Conference (MUC-6 and MUC-7) and the Automatic Content Extraction (ACE) competitions. In both cases, the particular genre represented is that of news reporting (including the different sub-genres of newswire and broadcast news in the case of ACE). Originally designed for evaluating coreference systems, these corpora have however also been used recently for anaphora resolution. In the following, we briefly discuss the composition and annotation schemes used in these corpora, as well as some of the problems (as noted for instance by (van Deemter and Kibble, 2000)).

MUC-6 and MUC-7 corpora

Successively released in 1995 and 1998, the MUC-6 and MUC-7 corpora both contain newspaper articles from the *Wall Street Journal*. All the annotated texts amount to about 65,000 words. The MUC-6 documents are exclusively about business related news (leadership changes, in particular), while MUC-7 documents are about plane crashes, space vehicles, and missile launches. The two datasets are divided into "dryrun" and "formal" documents, respectively used for training and testing: MUC-6 follows a 30/30 document split, while MUC-7 follows a 30/20 split.[8]

According to MUC-6 and MUC-7 annotation guidelines (Hirschman and Chinchor, 1998), coreference relationships can hold between elements of the following categories: proper names and named entities, Noun Phrases (including things like dates, currency

[7] These corpora are available through the Linguistic Data Consortium (LDC): `http://www.ldc.upenn.edu`.

[8] These figures correspond to the splits used during the official evaluation. Some 150 additional documents exist for MUC-6.

expressions, and percentages, but not conjoined NPs), bare nouns (including modifiers), and pronouns (including personal, possessive, and demonstrative pronouns, but not relative ones), referred collectively as *markables*. As for what constitutes a coreference relation, the guidelines go beyond so-called "basic coreference" to include bound anaphora, appositives, predicative nominals, and metonymies (among other things).

The coreference relations, along with the text layout (e.g., headline, location, sentence breaks), is encoded in an SGML format. A short excerpt from MUC-7 is reproduced in Figure 2.1. Markables that enter in a coreference relation are enclosed inside a pair of

```
<s>In <COREF ID="11" TYPE="IDENT" REF="12" MIN="quarter">the
third quarter</COREF>, <COREF ID="13" TYPE="IDENT" REF="10"
MIN="company"> the company, which is 61%-owned by Murphy Oil
Corp. of Arkansas, </COREF> had <COREF ID="100" MIN="loss">a
net loss of <COREF ID="17" TYPE="IDENT" REF="100">$46.9
million</COREF>, or <COREF ID="16" TYPE="IDENT" REF="17"
MIN="91 cents">91 cents a share</COREF>.</s>
```

Figure 2.1: An excerpt from the MUC-7 corpus

<COREF> and </COREF> tags. Crucially, only markables that are coreferential with other markables in the text are represented this way, which means that single-mention entities (i.e., singleton chains) are *not* encoded in the annotation. (An example is the NP *Murphy Oil Corp. of Arkansas* in the above excerpt.) Each coreferential markable is identified by a unique ID attribute. The additional REF attribute is used for markables whose referent has been previously introduced through another markable: the value of the REF attribute is the ID value of a coreferential markable (typically, that of its closest "antecedent"). In addition to these two attributes, a markable can also have a TYPE, MIN, STATUS attributes (see (Hirschman and Chinchor, 1998)) for details.

ACE corpus

The ACE corpus can be seen as a successor, as well as a refinement, of the MUC corpora. The overall goal of ACE is indeed broader, since it consists of the "detection and characterization of entities, relations, and events". The first ACE corpus was released in 1999, and has known various updates over the recent years. The discussion that follows describes only the 2002 Phase 2 release and focuses on the annotation pertaining to the "entity detection and tracking" task.

The ACE corpus has three parts, each containing around 75,000 words and corresponding to a different sub-genre: broadcast news (BNEWS), newspaper texts (NPAPER), and newswire texts (NWIRE). Each set is split into a `train` of 60,000 words part and a `devtest` part of 15,000 words. The precise split-up in terms of number of documents and mentions is shown in Figure 2.2.

	# documents		# mentions	
Dataset	`train`	`devtest`	`train`	`devtest`
BNEWS	216	51	10,086	2,608
NPAPER	76	17	11,410	2,504
NWIRE	130	29	10,868	2,630
ENTIRE ACE	422	97	32,364	7,742

Figure 2.2: Split-up of the ACE (Phase 2) corpus

There are two important differences from the MUC data. First, the annotated mentions in ACE are restricted to 5 entity types: FACility, GPE (geo-political entity), LOCation, ORGanization, PERson. Second, the ACE corpus also annotates single-mention entities (i.e., entities whose chain contain only one mention).

The annotation format of ACE is also different from that of MUC in that it relies on two distinct files for each document: an SGML file that marks up the raw text, and an XML file that marks up the different entities (and their mentions) in the text. A small excerpt of an ACE XML file is given in Figure 2.3. More specifically, the XML file is organized in terms of `entity` elements: each entity has an ID attribute and contains a `entity_type` element

```
<entity ID="9801.139-E28">
   <entity_type GENERIC="TRUE">PERSON</entity_type>
      <entity_mention TYPE="NOMINAL" ID="28-121">
        <extent>
          <charseq>
          <!-- string = "Americans" -->
             <start>1526</start><end>1534</end></charseq>
        </extent>
        <head>
          <charseq>
          <!-- string = "Americans" -->
             <start>1526</start><end>1534</end></charseq>
        </head>
      </entity_mention>
      <entity_mention TYPE="PRONOUN" ID="28-122">
        <extent>
          <charseq>
          <!-- string = "they" -->
             <start>1541</start><end>1544</end></charseq>
        </extent>
        <head>
          <charseq>
          <!-- string = "they" -->
             <start>1541</start><end>1544</end></charseq>
        </head>
      </entity_mention>
</entity>
```

Figure 2.3: An excerpt from the ACE (Phase 2) corpus

specifying one of the 5 ACE types, and a list of `entity_mention` elements. Each of these mentions also has an ID and TYPE attributes (the latter characterized the head word of the mention, and has three possible values: NAME, NOMINAL, PRONOUN). Finally, the connection to the raw text is ensured by the an `extent` sub-element that encodes the character offsets of each mention.

Linguistic objections

Although the creation MUC and ACE has been salutary to the research on reference resolution in setting standards for annotation and evaluation (see Section 2.4), some of annotation choices are somewhat debatable from a theoretical linguistic point of view. The most severe criticisms have been voiced by (van Deemter and Kibble, 2000).[9] We briefly review some of these objections.

The major problem with these annotation schemes is that they fail to capture any coherent notion of coreference: they indeed adopt a very stretched-out definition of coreference (one that encompasses that of anaphora), leading to certain semantic inconsistencies. First, these schemes include *non-referring* expressions in their coreference annotation, as in (2.2)a.:

(2.2) a. Whenever *a solution* emerged, we embraced *it*

b. *Every TV network* reported *its* profits.

c. *Henry Higgins*, who was formerly *sales director of Sudsy Soaps*, became *president of Dreamy Detergents*.

This is a case of (bound) anaphora (i.e., *it* is clearly anaphoric to *a solution*), but not of coreference. There can't be *co*reference, since there isn't reference in the first place.[10] Including instances of bound anaphora in coreference chains may lead to additional problems, as shown in (2.2)b.: by positing coreference between *Every TV network* and *it*, one wrongly predicts that the referent of *it* is the set of all TV networks.

Bound anaphora are not the only type of expressions whose referentiality is problematic. Predicative Noun Phrases are also problematic, as illustrated in example (2.2)c. The issue in this example concerns *intensionality*: relating the three underlined mentions

[9] These authors' criticisms are about the MUC coreference annotation scheme, but they carry over to that of ACE.

[10] Unless one regards coreference as a relation, not between actual (model) objects, but between abstract *discourse entities*.

by a coreference relation wrongly predict that the sales director of Sudsy Soaps and the president of Dreamy Detergents are the same person.

2.4.2 Evaluation metrics

In addition to annotated corpora, the proper evaluation of anaphora/coreference systems also requires adequate scoring methods. These constitute an important aspect of study in the field by providing a way to compare systems and thus also shaping new directions of research. Most of the evaluation work on reference resolution has been so-called *intrinsic* evaluation, that is evaluation against a gold standard (in the form of an annotated corpus such as MUC or ACE). A lot less attention has been given to *extrinsic* evaluation: that is, evaluation through the embedding of resolver into another application. The work of (Kehler, 1997) and (Morton, 1999) can however be regarded as attempts in that direction, since these authors study the impact of coreference resolution in the context of larger tasks such as IE and QA, respectively.

The following discussion is limited to the main scoring metrics developed for intrinsic evaluation.

Anaphora resolution

Anaphora resolution is defined as the task that of finding the correct antecedent for each anaphoric mention. This means that anaphora resolvers can be evaluated using a simple *accuracy* measure, as defined in (2.3).

$$\text{Accuracy} = \frac{|\text{correctly resolved anaphors}|}{|\text{all anaphors}|} \tag{2.3}$$

That is, the accuracy of an anaphora resolver for a given document D is expressed as the ratio between the number of anaphora for which the system finds the correct antecedent in D and the total number of anaphora in D.[11] The use of this particular metric comes with

[11] (Mitkov, 2002b) call this measure the *success rate* of the anaphora resolution algorithm.

two important assumptions. First, the resolver is only allowed to pick a *single* antecedent for each anaphor. To see why this is important, imagine the extreme case of a system that would pick all mentions in D as antecedents: such a trivial system would get a perfect accuracy score. Second, all the "true" anaphors are given to the system: that is, the only errors made by the system are resolution errors (i.e., errors in antecedent selection). Again, one can think of an extreme case, such as a system that would only resolve "easy" cases.

As just noted, the accuracy measure is useful for measuring the performance of a system in the resolution phase (basically, steps 4-7 of RESOLVE). Other metrics have however been proposed with the aim of evaluating the anaphora resolution system as a whole (with potential errors made in the preprocessing steps 1-3). As an illustration, (Baldwin, 1997) propose an evaluation metric in terms of recall and precision; the computation for each of these is given in (2.4) and (2.5), respectively.

$$\text{Recall}_{AR} = \frac{|\text{correctly resolved anaphors}|}{|\text{all anaphors}|} \tag{2.4}$$

$$\text{Precision}_{AR} = \frac{|\text{correctly resolved anaphors}|}{|\text{anaphors identified by the system}|} \tag{2.5}$$

Recall_{AR} is basically the same as the Accuracy measure above, but it is now coupled with Precision_{AR} measure, which computes the ratio between the number of correctly resolved anaphors divided by the number of anaphors identified by the system.

Coreference resolution

The evaluation of coreference resolution systems is slightly more delicate, since one has to consider the entire partition (i.e., the set of chains) produced by the system and determine how well it matches the gold standard partition. Applying the anaphora resolution metrics above would only give an imprecise way of evaluating a coreference system: these metrics score pairwise $\langle antecedent, anaphor \rangle$ decisions and would miss *implicit* links that are only present through transitive closure. For instance, a system that produces two links

$\{\langle A, B\rangle, \langle B, C\rangle\}$ would not get credit for the link $\langle A, C\rangle$.

Three different metrics have been proposed for evaluating coreference performance: the MUC metric (Vilain et al., 1995), the B^3 metric (Bagga and Baldwin, 1998), the CEAF metric (Luo, 2005). Common to these metrics is: (i) they operate by comparing, for each document, the set of chains $\mathcal{S}$ produced by the system against the "true" chains $\mathcal{T}$, and (ii) they report performance in terms of *recall* and *precision*.[12] There are however important differences in how each metric computes these scores, each producing a different bias.

MUC metric: a link-based evaluation The MUC metric directly relies on the notion of coreference *links* (i.e., pairs of mentions) for computing its scores. Recall and precision are indeed obtained by determining the number of links that are common to $\mathcal{S}$ and $\mathcal{T}$.[13] Specifically, recall is the ratio between the number of links that are common to $\mathcal{S}$ and $\mathcal{T}$ and the total number of links in the $\mathcal{T}$, whereas precision is the ratio between the number of links common to $\mathcal{S}$ and $\mathcal{T}$ and the total number of links in $\mathcal{S}$. In terms of errors made by the system, recall penalizes the *missing* links (i.e., the links present in $\mathcal{T}$ but not in $\mathcal{S}$), whereas precision penalizes the *spurious* links (i.e., the links present in $\mathcal{S}$ but not in $\mathcal{T}$).

Let us see concretely how these different numbers can be computed. Suppose that S is one of the chains composing $\mathcal{S}$, and T one of $\mathcal{T}$. First, note that the number of links in a chain S (respectively T) can be simply computed as $|S| - 1$ (respectively, $|T| - 1$). This is because chains are equivalence classes: one only needs of $n - 1$ links to connect all the elements of a chain with n elements. The computation of the total number of links in $\mathcal{S}$ (respectively $\mathcal{T}$) obtains straightforwardly as a simple summation over the constitutive chains. The number of links common to $\mathcal{S}$ and $\mathcal{T}$ can also be computed efficiently by taking the intersection between the different S and T. Following the same rationale as above, there are indeed $|S \cap T| - 1$ common links between S and T (provided the $S \cap T$ is not empty).

[12] As usual, f-score is obtained by taking the harmonic mean of recall and precision. That is: f-score $= \frac{2 \times Recall \times Precision}{Recall + Precision}$.

[13] Technically, these sets of chains have first to be computed by taking the reflexive, transitive closure over the pairs in the so-called *response* and *key* files.

The precise definitions for recall and precision are given in (2.6) and (2.7).[14]

$$\text{Recall}_{\text{MUC}} = \frac{\sum_{S \in \mathcal{S} \cap T \in \mathcal{T} \neq \emptyset} |S \cap T| - 1}{\sum_{T \in \mathcal{T}} |T| - 1} \tag{2.6}$$

$$\text{Precision}_{\text{MUC}} = \frac{\sum_{S \in \mathcal{S} \cap T \in \mathcal{T} \neq \emptyset} |S \cap T| - 1}{\sum_{S \in \mathcal{S}} |S| - 1} \tag{2.7}$$

By far the most widely used, the MUC metric has however a number of shortcomings (see (Bagga and Baldwin, 1998), (Popescu-Belis and Robba, 1998), (Luo, 2005)). The first problem is that it favors systems that create large chains (and therefore fewer entities). This bias can sometimes lead to situations where a trivial strategy receives a better score than an intuitively better system. Thus, a system that produces a single chain often achieves a perfect recall without always having severe degradation in precision (see example below); this tendency is especially true for documents that have "large" chains. This lenience with respect to large chains comes from the fact that MUC, in effect, only counts as errors the *miminum* number of links required to map two S and T chains onto another. For instance, two sets of chains $T = \{\{m_1, m_2, m_3, m_6\}, \{m_4, m_5, m_7\}\}$ and $S = \{\{m_1, m_2, m_3, m_4, m_5, m_6, m_7\}\}$ can be reunited by positing a single extra link (i.e., there is only one spurious link). A second, related problem with this metric is that it doesn't give any credit for separating singleton chains. Recall that single mention entities are simply absent from the MUC annotation scheme. It is actually unclear how the MUC metric would score these, since it relies on pairwise links: by definition, singleton chains contain no such link.

B^3 metric: a mention-based evaluation The B^3 metric was directly designed to address the MUC metric's shortcomings. While MUC is link-based, $B^3$is *mention-based*: both recall and precision scores are computed at the level of each mention. Let S_m be the system chain containing mention m, and T_m be the true chain containing m. The recall *for* m is

[14]The implementation proposed here is slightly different from, but equivalent, to that of (Vilain et al., 1995). Vilain *et al.* use an intermedidate partition function for computing the common links. See the paper for details.

calculated as the ratio between the number of common mentions in S_m and T_m (i.e., the mention in $|S_m \cap T_m|$) and the total number of mentions in T_m. Similarly, the precision *for* m is calculated as the ratio between the number of mention in $|S_m \cap T_m|$ and the total number of mentions in S_m.

The recall and precision scores for the document are then obtained by averaging over the individual recall and precision scores, as shown in (2.8) and (2.9).

$$\text{Recall}_{\text{B}^3} = \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} \frac{|S_m \cap T_m|}{|T_m|} \tag{2.8}$$

$$\text{Precision}_{\text{B}^3} = \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} \frac{|S_m \cap T_m|}{|S_m|} \tag{2.9}$$

It is easy to see that this new metric, by definition, solves the problems faced by MUC. First, B^3 has no difficulty in the scoring of singleton chains, since the metric is no longer based on pairwise links but on individual mentions. Second, large chains are no longer unjustly favored: the number of errors in a given chain are being compounded, since these errors affects the computation of each mention's score. Finally, note that the B^3 formulation provides some extra flexibility: although all errors receive the same weight in (2.8) and (2.9), one can potentially introduce different weights for different mentions.

CEAF: an entity-based evaluation Yet another evaluation strategy is proposed with The Constrained Entity Aligned F-Measure (CEAF) of (Luo, 2005). While MUC was link-based and B^3 was mention-based, this metric can be described as *entity-based*. The guiding principle of CEAF is that each entity should only used once in the evaluation of the entire partitioning. That is, each system chain S is mapped to *at most one* true chain T. This was neither the case with MUC or B^3, where each chain can in principle be used several times.

More concretely, this metric works by first computing the best of all possible one-to-one mappings, $G(\mathcal{S}, \mathcal{T})$, between the sets of chains $\mathcal{S}$ and $\mathcal{T}$. The best mapping, g^*, is the one that maximizes the total similarity, $\Phi(g)$ for a map g, which is just the sum

over the pairwise similarity $\phi(S_i, T_i)$ over pairs of aligned S_i and T_i chains. The pairwise similarity $\phi(S_i, T_i)$ is simply the number of common mentions to the two chains: that is, $\phi(S_i, T_i) = |S_i \cap T_i|$. The entity alignment problem, although potentially very hard (there is an exponential number of possible maps), is equivalent to finding the optimal alignment in a bipartite graph for which there are polynomial time algorithms.[15]

Once the best map is found, the recall and precision can be easily computed, as in (2.10) and (2.11).

$$\text{Recall}_{\text{CEAF}} = \frac{\Phi(g*)}{\sum_i \phi(T_i, T_i)} \tag{2.10}$$

$$\text{Precision}_{\text{CEAF}} = \frac{\Phi(g*)}{\sum_i \phi(S_i, S_i)} \tag{2.11}$$

In words, recall is computed as the ratio between total similarity for the best map $g*$ and the number of mentions in all the $\mathcal{T}$ (i.e., the self-similarity between each T of $\mathcal{T}$). Precision, on the other hand, is the ratio between the total similarity for $g*$ and the the number of mentions in $\mathcal{S}$(i.e., the self-similarity between each S of $\mathcal{S}$).

Clearly, CEAF also constitutes an improvement over the MUC metric. In particular, exceedingly large chains are strongly penalized by CEAF: each of them can indeed only be used once during evaluation. Note that CEAF is arguably the toughest metric, since it is the only metric within which predicting a valid coreference link might not receive any credit. For instance, imagine a case where a system predicts a chain $\{m_1, m_2, m_3, m_4, m_5, m_6\}$ but the true partition consists of two chains: $\{m_1, m_2, m_3\}$ and $\{m_4, m_5, m_6\}$. Under CEAF, the predicted chain can only be used once: that is, the predicted chain can only be mapped onto one of two true chains. The consequence is that two valid links will not receive any credit whatsoever. Overall, it remains unclear to us whether CEAF represents an improvement over B^3. In this dissertation, we will report the coreference scores in terms of the three metrics presented above.

[15](Luo, 2005) uses the so-called Kuhn-Munkres algorithm. See the paper for more details.

A simple example Table 2.2 provides a simple example illustrating how the three above coreference metrics operate. $\mathcal{T}$ is the set of true chains, $\mathcal{S}_1$ and $\mathcal{S}_2$ are the partitions produced by two hypothetical coreference resolvers.

$$\begin{array}{lcl} \mathcal{T} & = & \big\{\{m_1, m_3, m_5\}, \{m_2\}, \{m_4, m_6, m_7\}\big\} \\ \mathcal{S}_1 & = & \big\{\{m_1, m_2, m_3, m_6\}, \{m_4, m_5, m_7\}\big\} \\ \mathcal{S}_2 & = & \big\{\{m_1, m_2, m_3, m_4, m_5, m_6, m_7\}\big\} \end{array}$$

Table 2.2: Three hypothetical coreference partitions over 7 mentions

Recall (R), precision (P), and f-score (F) scores for the three metrics for this example are summarized in Table 2.3.

Metric	$\mathcal{S}_1$			$\mathcal{S}_2$		
	R	P	F	R	P	F
MUC	.50	.40	.44	1.0	.66	.79
B^3	.57	.42	.48	1.0	.39	.56
CEAF	.57	.57	.57	.43	.43	.43

Table 2.3: Comparative results between MUC, B^3, and CEAF

The bias of the MUC metric for large chains is shown by the fact that it gives better recall *and* precision scores for $\mathcal{S}_2$ even though this partition is totally uninformative. More intuitively, B^3 highly penalizes the precision of this partition: precision errors are here computed for each mention. CEAF is the harshest on $\mathcal{S}_2$, and in fact is the only metric that prefers $\mathcal{S}_1$ over $\mathcal{S}_2$. Finally, note that CEAF assigns the same recall and precision: this is because the two systems partitions the same set of mentions.

2.5 Summary

In this chapter, we gave an overview of: (i) the previous computational treatments to reference resolution, and (ii) the corpora and metrics used for evaluating the resolution systems. We started by presenting a generic algorithm, RESOLVE, that details the different steps involved in reference resolution, and that describes the majority of previous approaches.

The main trends of research were briefly described from an historical perspective, from knowledge-based to heuristic-based to learning-based systems. Using Soon et al. (2001) as an illustration, we describe the standard machine learning approach in more detail: as discussed, most existing learning-based systems recast the problem of anaphora/coreference resolution using a binary classifier. In the case of coreference resolution, this classifier is coupled with a link-selection algorithm that selects a single antecedent per each anaphor. In effect, this approach amounts to reducing the task of coreference resolution to that of anaphora resolution. We also discuss the main challenges faced by this standard approach (in particular, the potential inadequateness of both the classification model and the clustering algorithm, as well as the lack of adequate and reliable knowledge), and some recent work that addresses them.

Finally, we presented the various corpora annotated with coreference information (the main ones being the MUC-6-MUC-7 and the ACE corpus) and the different evaluation metrics proposed for anaphora resolution and coreference resolution (in particular, the MUC, the B^3, and the CEAF metrics).

Chapter 3

A ranking approach to pronoun resolution

In this chapter, we propose a maximum entropy ranking approach to pronoun resolution as an alternative to commonly used classification-based approaches. Classification approaches consider only one or two candidate antecedents for a pronoun at a time, whereas ranking allows all candidates to be evaluated together. We argue that this provides a more natural fit for the task and show that it also delivers important performance improvements. Tested on the ACE datasets, the ranker obtains error reductions ranging from 5.4% to 31% when compared to three previously proposed classifier-based approaches. Furthermore, we show the ranker offers some computational advantage over the best performing classifier-based approach, since it easily allows the inclusion of more candidate antecedents during training. This approach leads to a further error reduction of 8.1%.[1]

[1] This chapter is based on and extends (Denis and Baldridge, 2007a).

3.1 Maximum entropy models

A decisive step in applying machine learning techniques to a particular task is the identification of a well-defined learning problem. This problem has to: (i) adequately capture the structure of the task, and (ii) be suited to a particular learning algorithm. For many tasks, this often means classification. Mathematically well understood, classification problems have the advantage that they can be learned with a wide range of learning methods. Furthermore, numerous NLP tasks, from part-of-speech tagging to syntactic parsing to semantic role labelling, have been successfully modeled or decomposed in terms of classification problems. As explained in Chapter 2, anaphora and coreference resolution has also been cast in terms of classification. In this chapter, we develop an alternative view in which anaphora resolution (in particular, pronoun resolution) is cast in terms of ranking. As we will see, the ranking approach provides a more natural way to capture the structure of the task.

In this section, we first introduce classification and ranking in general terms, and show how each task can be formulated using log-linear (aka maximum entropy) models. In the light of this formal introduction, we then turn in the next section to a critical assessment of the classification-based approaches proposed in the literature and motivate our ranking approach.

3.1.1 Classification

In a classification problem, one seeks to learn a function $cl : \mathcal{X} \rightarrow \mathcal{Y}$, which maps an input $x \in \mathcal{X}$ to a predefined class label $y \in \mathcal{Y}$. The determination of each input x's label is based on a vector of m features describing an input and a label $\bar{f} = \langle f_1(x, y), \ldots, f_m(x, y) \rangle$, where $f_j : \mathcal{X} \times \mathcal{Y} \rightarrow \Re$, and an associated vector of m parameters (i.e., weights) $\bar{w} =$

$\langle w_1, \ldots, w_m \rangle$, where $w_j \in \Re$, that have been learned during training.[2] In *linear* classification, this is accomplished by summing over the different weights:

$$cl(x) = \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \sum_{j=1}^{m} w_j f_j(x, y) \tag{3.2}$$

The training data in the classification scheme takes the form of a set of n examples $\mathcal{T}_{cl} = \{(x_i, y_i)\}_{i=1}^{n}$, where each input x_i has been annotated with its correct class y_i.

The task of learning is that of finding the optimal set of parameters given the set of training examples: that is, we want to learn the set of parameters that maximize the likelihood of the training data. A common learning technique is to use maximum entropy models, also known as log-linear or exponential models (e.g., (Berger et al., 1996)). Widely used within NLP, these discriminative models have some important advantages: (i) their focus is on directly modelling a discriminative function rather than on the probabilities of the observations, and (ii) they make it easier to incorporate many information sources without making independence assumptions. Crucial to our concern, these models can also be used for ranking, as we will see shortly.[3] Detailed presentations of these models are given in (Berger et al., 1996; Ratnaparkhi, 1998). The following discussion is inspired from and borrows the notation of (Collins and Koo, 2005).

In MaxEnt models, the parameters $\bar{w}$ are used to define a conditional probability,

[2] For instance, a generic binary feature takes the following form:

$$f_{p,y'}(x, y) = \begin{cases} 1 & if \text{ y = y' and p(x) = true} \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

where p is known as a contextual predicate.

[3] Note that other learning algorithms, such as perceptrons and support vector machines (SVMs), can also be used to learn classifiers *and* rankers.

which takes the following exponential form:

$$P_{\bar{w}}(y|x) = \frac{\exp \sum_{j=1}^{m} w_j f_j(x,y)}{\sum_{y' \in \mathcal{Y}} \exp \sum_{j=1}^{m} w_j f_j(x,y')} \tag{3.3}$$

The task of learning is defined as that of finding the set of parameters $\bar{w}$ that maximize the log-likelihood of the training data $\mathcal{T}_{cl}$.[4] Equivalently, this can be formulated as minimizing the following loss function with respect to $\mathcal{T}_{cl}$:

$$\begin{aligned} L_{\mathcal{T}_{cl}}(\bar{w}) &= \sum_{i=1}^{n} -\log P_{\bar{w}}(y_i|x_i) \\ &= \sum_{i=1}^{n} -\log \frac{\exp \sum_{j=1}^{m} w_j f_j(x_i,y_i)}{\sum_{y \in \mathcal{Y}} \exp \sum_{j=1}^{m} w_j f_j(x_i,y)} \\ &= \sum_{i=1}^{n} -\log \frac{1}{1+\sum_{y \neq y_i \in \mathcal{Y}} \exp -(\sum_{j=1}^{m} w_j f_j(x_i,y_i) - \sum_{j=1}^{m} w_j f_j(x_i,y))} \\ &= \sum_{i=1}^{n} \log \left(1 + \sum_{y \neq y_i \in \mathcal{Y}} \exp -(\sum_{j=1}^{m} w_j f_j(x_i,y_i) - \sum_{j=1}^{m} w_j f_j(x_i,y)) \right) \end{aligned} \tag{3.4}$$

As shown in this final equality of the objective function,[5] the goal of estimation is to find the set of parameters that maximizes *for each input* x_i from $\mathcal{T}_{cl}$ the following margin M_{cl} between the correct class y_i and the incorrect ones y:

$$M_{cl} = \sum_{j=1}^{m} w_j f_j(x_i,y_i) - \sum_{j=1}^{m} w_j f_j(x_i,y) \tag{3.5}$$

Intuitively, this means that the goal of estimation is to increase the weights of the features that predict the correct class x_i and to decrease those of the features predicting the other

[4] The relation to the concept of maximum entropy is the following: the model that maximizes the likelihood of the training data is also the model that maximizes the entropy over the set of models consistent with the empirical observations on the training data (Berger et al., 1996).

[5] The various manipulations in (3.4) follow for the most part from the definitions of logarithm and exponentiation.

classes x_j.

Before turning to ranking, note that the objective function above is slightly incomplete. Given the observed tendency of log-linear models to over-fit the training data (especially with sparse data), one often incorporates a regularization term in the objective function. Typically, this is done by using a Gaussian prior on the weights which has the effect of penalizing extreme values (Chen and Rosenfeld, 1999). That is, the actual loss function in (3.4) should really be:

$$L_{\mathcal{T}_{cl}}(\bar{w}) = \sum_{i=1}^{n} -\log P_{\bar{w}}(y_i|x_i) + \sum_{j=1}^{m} \frac{w_i^2}{2\sigma_i^2} \tag{3.6}$$

Finally, note that different algorithms for effectively estimating parameters have been proposed (see (Malouf, 2002) for a comparison); in this dissertation, we used the *limited memory variable metric* optimization method implemented in the Toolkit for Advanced Discriminative Modeling[6].

3.1.2 Ranking

While numerous NLP problems have been cast as classification, others have been cast as (re)ranking problems. A common example is parse selection. (e.g., (Collins and Duffy, 2002; Charniak and Johnson, 2005; Osborne and Baldridge, 2004; Toutanova et al., 2004)).[7] In parse selection, one must identify the best analysis out of some set of parses produced by a grammar. Different sentences of course produce very different parses and very different numbers of parses, depending on the ambiguity of the grammar. To our knowledge, classification has never been explored for this problem. Other uses of rankers involve question-answering (Ravichandran et al., 2003) and tactical generation (Velldal and Oepen, 2006). Common to these different problems is that one is concerned with the identification of a

[6] Available from `tadm.sf.net`.

[7] A *re*ranker is ranker that is applied to the output of a previous model used to produce a n-best list of candidates.

single "best" candidate among a set of possible candidates.

More formally, the goal in ranking is to learn a scoring function $rk : \mathcal{Y}(x) \rightarrow \Re$, which maps a candidate $y \in \mathcal{Y}(x)$ for a given input x to a *score*. For instance, y might be one candidate among a set of parses $\mathcal{Y}(x)$ for a sentence x. By assigning a score to each candidate x_i, this function defines a total ordering over the entire candidate set $\mathcal{Y}(x)$. As for classification, one computes this score based on a set of weighted features, with the difference that features are now defined solely based on the candidate (instead on being based on input-label pair): that is, features are $f_j : \mathcal{Y}(x) \rightarrow \Re$. The score assigned to each candidate y is computed as follows:

$$rk(y) = \sum_{j=1}^{m} w_j f_j(y) \tag{3.7}$$

Given a set of candidates $\mathcal{Y}(x)$, the most likely candidate $\widehat{y}$ is simply the one that gets the highest score:

$$\begin{aligned} \widehat{y} &= \operatorname*{argmax}_{y \in \mathcal{Y}(x)} rk(y) \\ &= \operatorname*{argmax}_{y \in \mathcal{Y}(x)} \sum_{j=1}^{m} w_j f_j(y) \end{aligned} \tag{3.8}$$

In ranking, the training data $\mathcal{T}_{rk} = \{(x_i, \mathcal{Y}(x_i), y_i^*)\}_{i=1}^{n}$ is a set of tuples where each object x_i is associated with a set of candidates $\mathcal{Y}(x_i)$ among which one candidate y_i^* is singled out as the correct candidate.[8]

In MaxEnt models, the conditional probability of y being the correct candidate for an input x takes the following exponential form:

$$P_{\bar{w}}(y|x) = \frac{\exp \sum_{j=1}^{m} w_j f_j(y)}{\sum_{y' \in \mathcal{Y}(x)} \exp \sum_{j=1}^{m} w_j f_j(y')} \tag{3.9}$$

[8]For ease of exposition, we restrict the discussion to the case in which there is a unique correct candidate, but this is by no means a requirement (that is, one can have various correct candidates).

The objective function for ranking takes the following form:

$$\begin{aligned} L_{\mathcal{T}_{rk}}(\bar{w}) &= \sum_{i=1}^{n} -\log P_{\bar{w}}(y^*|x) \\ &= \sum_{i=1}^{n} -\log \frac{\exp \sum_{j=1}^{m} w_j f_j(y)}{\sum_{y' \in \mathcal{Y}(x)} \exp \sum_{j=1}^{m} w_j f_j(y')} \end{aligned} \tag{3.10}$$

Using the same manipulations as in (3.4):

$$L_{\mathcal{T}_{rk}}(\bar{w}) = \sum_{i=1}^{n} \log \left(1 + \sum_{y \neq y* \in \mathcal{Y}(x)} \exp - (\sum_{j=1}^{m} w_j f_j(y^*) - \sum_{j=1}^{m} w_j f_j(y)) \right) \tag{3.11}$$

That is, the goal of estimation is here to find the set of parameters that maximizes *for each set of candidates* $\mathcal{Y}(x)$ the following margin M_{rk} between the correct candidate y^* and the incorrect ones y:

$$M_{rk} = \sum_{j=1}^{m} w_j f_j(y^*) - \sum_{j=1}^{m} w_j f_j(y) \tag{3.12}$$

That is, one seeks for each candidate set the parameters that best teases the correct candidate y^* apart from all the other candidates y.

3.2 Modeling pronoun resolution

We now turn to the actual modeling of the pronoun resolution task. As introduced in Chapter 1, the task of anaphora resolution —of which pronoun resolution is an instance— boils down the process of selecting the correct antecedent for each anaphor in a given document. More specifically, this process is a function $\sigma : \mathcal{A} \rightarrow \mathcal{C}_\pi$ which takes as input an anaphoric pronoun $\pi \in \mathcal{A}$ and a set of possible antecedent candidates $\mathcal{C}_\pi = \{\alpha_1, \ldots, \alpha_n\}$, and outputs one of the candidates $\widehat{\alpha} \in \mathcal{C}_\pi$.[9] Since $\mathcal{A}$ and $\mathcal{C}_\pi$ are both subsets of the set of mentions $\mathcal{M}$ in the document, σ is in fact a partial function over $\mathcal{M}$. Typically, the candidate set $\mathcal{C}_\pi$

[9]This formulation makes the assumption that the resolution of an anaphor is independent from the resolutions of other anaphors, which is of course incorrect. This issue will be directly tackled in the treatment we give to coreference resolution in Chapter 5.

is assumed to be the mentions that linearly precede the anaphor π.

3.2.1 Antecedent selection with classification

Given the description of classification in the previous section, it is easy to see that trying to cast antecedent selection in terms of classification faces some important challenges. A naive approach would be to regard the different anaphors $\pi \in \mathcal{A}$ as inputs and the different antecedent candidates $\alpha_i \in \mathcal{C}_\pi$ as class labels. Under this approach, antecedent selection would simply equal class assignment. This approach is however not tenable in practice, since the number of classes would be prohibitively large (leading to important sparsity issues) and will vary considerably from one anaphor to the other (classification problems traditionally use a stable set of class labels).

Given that antecedent selection doesn't directly lend itself to classification, researchers have investigated ways to *coerce* this task into a classification problem. We discuss two approaches presented in the literature: the Single-Candidate Classifier and the Twin-Candidate Classifier. Common to both approaches is that antecedent selection is broken down into separate binary classification decisions, which are then used to impose a ranking on the candidate set.

The Single-Candidate Classifier

As discussed in Chapter 2, the most common approach has been to recast the task as a pairwise binary classification problem (e.g., (Morton, 2000; Kehler et al., 2004a)). Under this approach, a classifier maps pronoun-candidate pairs, $\langle \pi, \alpha_i \rangle \in \mathcal{M} \times \mathcal{M}$, into one of two class labels: COREF or $\neg$COREF. Viewed in probabilistic terms, we model $P_{scc}(c|\langle \pi, \alpha_i \rangle)$, where $c \in \{\text{COREF}, \neg\text{COREF}\}$. The corresponding exponential model is as follows:

$$P_{scc}(\text{COREF}|\langle \pi, \alpha_i \rangle) = \frac{\exp \sum_{j=1}^{n} w_j f_j(\langle \pi, \alpha_i \rangle, \text{COREF})}{\sum_{c} \exp \sum_{j=1}^{n} w_j f_j(\langle \pi, \alpha_i \rangle, c)} \tag{3.13}$$

In effect, the classifier determines for each candidate α_i whether α_i is (or is not) a "good" antecedent with respect to the anaphoric pronoun π.

Note that under this approach, the simple application of the model doesn't guarantee that an antecedent is predicted for each anaphor. Thus, there will be cases in which the model classifies several candidates as COREF, and cases in which no candidate will be classified as COREF. This means that an extra step is required to effectively select *one* antecedent. An obvious approach here is to compare the scores of the different candidates and pick as antecedent the one that receives the highest score w.r.t. to the COREF class. This decision rule is given in (3.14).

$$\widehat{\alpha} = \underset{\alpha_i \in \mathcal{C}_\pi}{\operatorname{argmax}} \sum_{j=1}^{m} w_j f_j(\langle \pi, \alpha_i \rangle, \text{COREF}) \tag{3.14}$$

This comparison is problematic however since the probabilities outputed by the single-candidate model are indirect, potentially imperfect, estimates of the true candidate probabilities w.r.t. to antecedent selection. This comes from the fact that during training the different antecedent candidates are never compared. The different candidates for the a given pronoun are considered *independently*, since only a *single* candidate is evaluated at a time. Each pronoun-candidate pair is indeed modeled as a separate event: given an anaphor π and a set of candidates $\mathcal{C}_\pi$, there are $|\mathcal{C}_\pi|$ distinct events pairs (i.e., $|\mathcal{C}_\pi|$ events). The contribution of each feature during training is determined based on how well this feature predicts the two possible classes, instead of being determined based on how well it helps tease apart the actual antecedent from the non-antecedents. For instance, assuming a correct antecedent α_i for the pronoun π, we are here trying to maximize for the following margin:

$$M_{scc} = \sum_{j=1}^{m} w_j f_j(\langle \pi, \alpha_i \rangle, \text{COREF}) - \sum_{j=1}^{m} w_j f_j(\langle \pi, \alpha_i \rangle, \neg\text{COREF}) \tag{3.15}$$

The problem is that we could potentially have situations in which a given feature f_1 is assigned a bigger weight than a second feature f_2 despite of f_2 being more discriminating

than f_1 wrt antecedent selection. Consider for instance the toy example in Table 3.2.1: this example has only three anaphors π_1, π_2, and π_3, each with different candidate sets (among which is the correct antecedent), and two features: f_1 and f_2. Since each antecedent

Anaphor	Candidate Set	Class	Feature vector
π_1	$\alpha_{\pi_1,1}$	$\neg$COREF	f_1
	$\alpha_{\pi_1,2}$	COREF	f_2
π_2	$\alpha_{\pi_2,1}$	$\neg$COREF	f_1
	$\alpha_{\pi_2,2}$	COREF	f_2
π_3	$\alpha_{\pi_3,1}$	COREF	f_1
	$\alpha_{\pi_3,2}$	$\neg$COREF	f_2
	$\alpha_{\pi_3,3}$	$\neg$COREF	f_2
	$\alpha_{\pi_3,4}$	$\neg$COREF	f_2
	$\alpha_{\pi_3,5}$	$\neg$COREF	f_2
	$\alpha_{\pi_3,6}$	$\neg$COREF	f_2

Table 3.1: Instances for pairwise binary classification

generates a distinct event, we have 10 different events overall (3 positive, and 7 negative) for this example. The features are distributed as follows: f_1 is associated with a positive instance 1 out of 3 times, while f_2 is associated with a positive instance 2 out of 7 times. This means that f_1 is likely to receive a larger weight than f_2 w.r.t. to the COREF class, even though f_2 predicts the right antecedent more reliably than f_1 (in 2 out of 3 cases).

The Twin-Candidate Classifier

To overcome the deficiencies of the single-candidate model, (Yang et al., 2003) propose a model in which *pairs* of candidate antecedents are considered: the so-called *twin-candidate* model.[10] In this approach, classification is still binary (the labels now represent the two candidates being compared), but the probabilities are now conditioned on the anaphoric pronoun π and a *pair* of candidates $\langle \alpha_i, \alpha_k \rangle \in \mathcal{M} \times \mathcal{M}$. Concretely, the model takes the

[10]While the twin-candidate approach is often associated with the work of X. Yang, the idea of using pairs of candidates actually originates in (Connolly et al., 1997).

following form: $P_{tcc}(c|\langle\pi, \alpha_i, \alpha_k\rangle)$, and can be given the following exponential form:

$$P_{tcc}(\text{FIRST}|\langle\pi, \alpha_i, \alpha_k\rangle) = \frac{\exp \sum_{j=1}^{n} w_j f_j(\langle\pi, \alpha_i, \alpha_k\rangle, \text{FIRST})}{\sum_c \exp \sum_{i=1}^{n} w_j f_j(\langle\pi, \alpha_i, \alpha_k\rangle, c)} \tag{3.16}$$

Here, c ranges over the two classes $\{\text{FIRST}, \text{SECOND}\}$ which correspond to choosing α_i or α_k, respectively, as a "better" antecedent with respect to the anaphoric pronoun π.

During training, each triple contains: (i) the anaphor, (ii) an *antecedent* mention, and (iii) a *non-antecedent* mention. Instances are labelled as FIRST or SECOND depending on whether the antecedent comes either before or after the non-antecedent in the text, respectively. The model induced through training is a preference model between any two candidates: as with the single-candidate classifier, the simple application of the model doesn't yet yield a predicted antecedent. That is, finding the final antecedent requires an extra step. (Yang et al., 2003; Yang, 2005) use a round-robin algorithm, which works by comparing all candidates in a pairwise fashion, and picking as the antecedent the one that accumulates the most victories.[11] This is captured in the decision rule in (3.17):

$$\widehat{\alpha} = \underset{\alpha_i \in \mathcal{C}_\pi}{\text{argmax}} \sum_{\alpha_k \neq \alpha_i \in \mathcal{C}_\pi} \alpha_i \succ \alpha_k \tag{3.17}$$

where:

$$\alpha_i \succ \alpha_k = \begin{cases} 1 & \text{if } \sum_{j=1}^{m} w_j f_j(\langle\pi, \alpha_i, \alpha_k\rangle, \text{FIRST}) > \sum_{j=1}^{m} w_j f_j(\langle\pi, \alpha_i, \alpha_k\rangle, \text{SECOND}) \\ & \text{or } \sum_{j=1}^{m} w_j f_j(\langle\pi, \alpha_k, \alpha_i\rangle, \text{SECOND}) > \sum_{j=1}^{m} w_j f_j(\langle\pi, \alpha_k, \alpha_i\rangle, \text{FIRST}) \\ 0 & \text{otherwise} \end{cases} \tag{3.18}$$

Note that the twin-candidate approach is computationally much more intensive than the

[11] (Connolly et al., 1997) instead uses a greedier procedure by which each pairwise comparison results in the elimination of the loser candidate.

single-candidate approach, both during the training and the application of the model. Each pronoun indeed potentially generates $|\mathcal{C}_\pi|^2$ distinct events (instead of just $|\mathcal{C}_\pi|$ for the single-candidate classifier). That is, the total complexity is now *cubic* in the number of mentions in the document; it was only *square* in the case of the single-candidate classifier. This is potentially an important drawback, especially with documents that contain a large number of mentions.

On paper, the twin-candidate classifier seems to be a better alternative than the single-candidate classifier: its main advantage is to make the competition between pairs of candidates part of the training criterion. That is, this model directly captures the *relative* goodness of different antecedent candidates for the same pronoun. From this point of view, this approach is similar to error-correcting output coding (Dietterich, 2000), an ensemble learning technique which is especially useful when the number of output classes is large. It can thus be seen as a group of models that are individual experts on teasing apart two different candidates. Nonetheless, this approach is still hampered by the fact that this model's probability estimates are only based on *two* candidates rather than all that are available. This means that unjustified independence assumptions are still made during model training and usage potentially hurting performance. In particular, it is –incorrectly– assumed in this model that the preference between a candidate α_i and α_j is independent of the preference between α_i and any other candidate α_k. As just noted, another potential problem for this approach is its computational cost.

3.2.2 Antecedent selection as ranking

While the twin-candidate strategy is an improvement over the single-candidate approach, it does not address the fundamental problem that pronoun resolution is not characterized optimally as a classification task. The nature of the problem is in fact much more like that of parse selection. Thus, we can view a text as presenting us with different analyses (candidate antecedents) which each pronoun could be resolved to.

Under the ranking approach, one is directly estimating the probability $P(\alpha_i|\pi)$, which is the probability of α_i being the "best" antecedent for the pronoun π.

$$P_{rk}(\alpha_i|\pi) = \frac{\exp \sum_{j=1}^{m} w_j f_j(\pi, \alpha_i)}{\sum_k \exp \sum_{j=1}^{m} w_j f_j(\pi, \alpha_k)} \tag{3.19}$$

The advantage of the ranker lies in the fact that it compares *all* the candidates at once, rather than in a piecemeal fashion. From that perspective, the ranker can be seen as a generalization over the twin-candidate classifier. The crucial point is that the comparison is part of the training criterion: each candidate α_i for a pronoun π is assigned a score with respect to the entire candidate set $\mathcal{C}_\pi$. Recall from the previous section that in ranking the parameters are adjusted in a way that maximizes the margin between the correct candidate and the bad candidates. In the present case, this means, for each anaphoric pronoun π, maximizing the margin between the antecedent α^* and the non-antecedents α:

$$M_{rk} = \sum_{j=1}^{m} w_j f_j(\alpha^*) - \sum_{j=1}^{m} w_j f_j(\alpha) \tag{3.20}$$

Once the parameters have been estimated, determining the "best" candidate $\widehat{\alpha}$ is simply performed by picking the candidate in $\mathcal{C}_\pi$ that has the highest score.

$$\widehat{\alpha} = \underset{\alpha_i \in \mathcal{C}_\pi}{\operatorname{argmax}} \sum_{j=1}^{m} w_j f_j(\alpha_i) \tag{3.21}$$

Given that the comparison between different candidates is directly part of the training criterion, we know that the score received by a candidate α_i is a true estimate of how well α_i fares against all the other candidates w.r.t. to being the best antecedent.

Another potential advantage of the ranking approach lies in the fact that features simply are the contextual predicates instead of being the combination of a contextual predicate combined with a class label, as is the case with classification. This has two impli-

cations. First, the total number of features is much smaller than with the classifiers (half the number of the features of the Single-Candidate Classifier and a fourth of the number of the features of Twin-Candidate Classifier).[12] Second, features can now be shared across different outcomes. This sharing is part of what makes rankers work well for tasks that cannot be easily cast in terms of classification: features are not split across multiple classes and instead receive their weights based on how well they predict correct outputs rather than correct labels.

3.3 Implemented systems

In the following, we compare four different pronoun resolvers: the first three systems are reimplemented versions of systems that have been proposed in the literature. In particular, we implemented two versions of the single-candidate classifier as found in (Kehler et al., 2004a) and (Yang, 2005), respectively. For the twin-candidate classification system, we followed the approach of (Yang, 2005). The implementation of the ranking system is entirely new.

Since the probability models used for the different models have been described in the previous section, we focus here on the different training and testing procedures. Before describing each individual system, note that all systems were developed, trained and tested on the ACE corpus, that is a corpus originally annotated with coreference chains. This means that in principle, an anaphoric pronoun can have several antecedents. In order to guide learning toward the mention that is the most likely to have caused the pronominalization, we take the *closest antecedent* as the only true antecedent: all coreferential mentions except the closest were eliminated before training. The different systems were tested on the ACE corpus: true mention boundaries from the corpus were assumed.

[12]The Twin-Candidate Classifier produces twice the number of features of the Single-Candidate Classifier, since it creates distinct features for each of the two candidates being compared.

3.3.1 Single-candidate classifiers

For the two single-candidate classifiers, we use training and test procedures proposed in (Yang, 2005)[13] and in (Kehler et al., 2004a), respectively. These two implementations differ in terms of both the way they sample the training instances, and in the way they select the antecedent candidate set during testing.

Training

Training instances are constructed based on pairs of mentions of the form $\langle \pi, \alpha_i \rangle$, where π and α_i are the descriptions for an anaphoric pronoun and one of its candidate antecedents, respectively. Each such pair is assigned either a label COREF (i.e. a positive instance) or a label NOT-COREF (i.e. a negative instance) depending on whether or not the two mentions are marked as coreferential. The number of instances thus created is at worst square in the number of mentions in the document (if one assumes that all mentions preceding a pronoun are potential candidates).

Both systems coincide in the way they produce the positive instances: these are created for each anaphor π by selecting the closest *antecedent* α_i. They diverge however in the way they produce the negative instances. In (Yang, 2005), negative instances are created for each non-antecedent α_j that intervenes between α_i and π.[14] (Kehler et al., 2004a) instead propose to generate negative instances for *all* non-antecedents that precede the anaphor.

Resolution

Once trained, the classifier is used to select a unique antecedent for each anaphoric pronoun in the test documents. This is done in two steps. First, each pronoun π is paired with each mention α_i in the candidate set $\mathcal{C}_\pi$, and the instance thus created in submitted to the

[13] This first model is the baseline used by (Yang, 2005) to evaluate his Twin-Candidate model.

[14] As discussed in Chapter 2, this way of selecting training instances is that of (Soon et al., 2001).

classifier. Second, the antecedent candidate $\widehat{\alpha}$ which receives the highest score w.r.t. to the COREF class is selected as the correct antecedent.[15] The two implementations of the single-candidate classifiers differ in the way they define the candidate set, in a way that is consistent with the way they each sample the training data. In (Yang, 2005), $\mathcal{C}_\pi$ contains only the mentions that appear in a window of 3 sentences from the anaphor π: this is motivated by the fact that pronouns show a strong tendency to take very local antecedents. (Kehler et al., 2004a), by contrast, consider all mentions that precede the anaphor π.

3.3.2 Twin-candidate classifier

The twin-candidate model was first proposed by Yang et al. (Yang et al., 2003) in the context of coreference resolution. (Yang, 2005) and Ng (Ng, 2005a) more recently used it specifically for the pronoun resolution task. In the following, we describe the training and testing procedures of (Yang, 2005).

Training

Training instances are constructed based on *triples* of mentions of the form $\langle \pi, \alpha_i, \alpha_j \rangle$, where π describes a pronominal anaphor and α_i and α_j are the descriptions for two of its candidate antecedents and α_i is stipulated to be closer to π than α_j. These instances are labeled either FIRST if α_i is the correct antecedent or SECOND if α_j is the correct antecedent. For this to work, one has to add an additional constraint on the creation of instances, namely: exactly one and only one of the two candidates can be coreferential with the pronoun. As we already pointed out, the number of instances created is much larger than with the single-candidate classifier: it is now cubic in the number of mentions in the document. In order to obviate this problem, (Yang, 2005) suggests restricting the set of candidate set to a window of 3 sentences including the sentence of the pronoun, and the immediately preceding two

[15] Note that this score didn't always reach a probability for the COREF class of over .5. Concretely, this means that the use of the standard link-selection techniques (as described in Chapter 2) would have resulted in some anaphors not being resolved.

sentences.

Resolution

Once trained, the twin-candidate classifier is used to select a unique antecedent for the given anaphoric pronoun π. Like Yang et al. (Yang, 2005) and Ng (Ng, 2005a), we use a round robin algorithm to compare the members of the candidate set for π. More specifically, test instances are created for each pair of candidates, α_i and α_j, where α_j precedes α_i. These instances are presented to the classifier, which determines which one of the candidates is preferred; the winner of the comparison gets one point. Finally, the candidate with the most points at the termination of the round robin competition gets selected as the antecedent for π. Following (Yang, 2005), we use a window of 3 sentences as was done in training.

3.3.3 Ranker

The following describes our training and resolution procedures for the ranking system.

Training

The training instances for the ranker system are built based on an anaphoric pronoun π and the set of its antecedent candidates $\mathcal{C}_\pi$. The candidate set is composed of: (i) the closest antecedent for π, which is singled out as such, and (ii) a set of non-antecedents. The construction of the latter set proceeds by taking the closest antecedent as an anchor and adding all the non-antecedents that occur in a window of 3 sentences around it (including the current sentence of the antecedent, the preceding sentence, and the two following sentences).

Resolution

Once trained, the ranker is used to select a unique antecedent for each anaphoric pronoun. We build our candidate set in the same way as was done for the twin-candidate model: that is, by considering the preceding mentions that occur in a window of 3 sentences, including

the pronoun's sentence and the 2 sentences preceding it. The selection of the antecedent is straightforward with the ranker, since it simply boils down to picking the candidate for which the model outputs the highest score.

3.4 Feature set

This section describes the feature set used in the different systems. Although this won't be made explicit in the description below, recall from Section 3.2 that features are different objects for classifiers and rankers. Also, note that in the twin-candidate model, each feature describing a candidate will in fact give rise to two distinct features, corresponding to each of the two candidates being compared.

Our focus in feature design was to capture linguistically relevant information, while relying on very limited linguistic processing. In particular, we only made use of a sentence detector, a tokenizer, a POS tagger (as provided by the OpenNLP Toolkit[16]) and the Wordnet database[17]. Recall that we assume the mention boundaries as given by the corpus.

The features were hand-selected and they fall into five main categories, which are developed below. Roughly, all of these features describe properties of either the antecedent candidate, or the relation between the anaphor and the candidate. The detailed feature set is summarized in table 3.2.

Linguistic form: This includes features pertaining to the referential form of the antecedent candidate: in particular, whether it is a proper name, a definite description, an indefinite NP, or a pronoun.

Context: This includes features describing the context of the antecedent candidate: these features can be seen as approximations of the grammatical roles, and as such inform us on the salience of the potential candidate (Grosz et al., 1995). For instance, we

[16] Available from `opennlp.sf.net`.
[17] `http://wordnet.princeton.edu/`

include as features the part of speech tags surrounding the candidate, as well as a feature that indicates whether the potential antecedent is the first mention in a sentence (approximating subject-hood), and a feature indicating whether the candidate is embedded inside another mention.

Distance: This includes features capturing the distance between the anaphor and the potential antecedent: pronouns due to their lack of lexical meaning are known to favor antecedents that are close-by (e.g., (Ariel, 1988; McEnery et al., 1997)). More specifically, we measured distance both in terms of the number of sentences and mentions intervening between them. Binned values were used for these different distance measures.

Morphosyntactic agreement: This includes features that encode the gender, number, and person of the two mentions. These are determined for non-pronominal NPs using heuristics based on the part of speech tags (e.g., `NN` vs. `NNS` for number) and the actual strings of the mentions (e.g., whether the mention contains a male/female first name or honorific for gender). These features take the form of pairs of attributes, making sure that not only strict agreement (e.g., *singular-singular*) but also mere compatibility (e.g., *masculine-unknown*) is captured.

Semantic compatibility: This includes features designed to assess whether the two mentions are semantically compatible. For these features, we use the Wordnet database: in particular, we collected pairs of Wordnet senses from the synonym set (or synset) as well as from the synset of the direct hypernyms of this synset associated with each mention. In the case of common nouns, we used the synset associated with the first sense associated with the mention's head word. In the case of proper names, we used the synset associated with the name if available, and the string itself otherwise. For pronouns (which are not part of Wordnet), we simply used the pronominal form.[18]

[18]This strategy produces a large number of potentially sparse features, but we find it to work better than using similarity measures developed for Wordnet (e.g., Pedersen et al. (2004)).

In addition to the simple features described above, we design composite features, combining distances and the type of the pronoun (e.g., reflexive, possessive).

Linguistic Form	
pn	α is a proper name {1,0}
def_np	α is a definite description {1,0}
indef_np	α is an indefinite description {1,0}
pro	α is a pronoun {1,0}
Context	
left_pos	POS of the token preceding α
right_pos	POS of the token following α
surr_pos	pair of POS for the tokens surrounding α
Distance	
s_dist	Binned values for sentence distance between π and α
np_dist	Binned values for mention distance between π and α
Morphosyntactic Agreement	
gender	pairs of attributes {masc, fem, neut, unk} for π and α
number	pairs of attributes {sg, pl} for π and α
person	pairs of attributes {1, 2, 3, 4, 5, 6} for π and α
Semantic compatibility	
wn_sense	pairs of Wordnet senses for π and α

Table 3.2: Feature selection for pronoun resolution

3.5 Experiments and results

3.5.1 Corpus and evaluation

The training and testing datasets used for our experiments come from the ACE corpus, as described in Chapter 2. The `devtest` material was only used once, namely for final testing. Progress evaluation (including the estimation of the best regularization priors) during the development phase was done solely by jackknifing the training set (we used a 5-fold).[19]

[19]For each model, we tried the following regularization priors: 0,1,2,4,5,10,100,1000,10000,100000. All models except Kehler et al.'s reimplementation of the single-candidate classifier benefited from Gaussian smoothing. This is in accordance with what (Kehler et al., 2004a) found.

In our experiments, we used all forms of third person pronoun (including reflexive and possessive forms) that were annotated as ACE "markables". This excludes pleonastics and references to eventualities or to non-ACE entities (that is, mentions that didn't fall into one of the five entity types used in ACE). Together, the three ACE datasets contain $4,389$ and $1,093$ referential pronouns, for training and testing, respectively.

Also, note that in building our antecedent candidate sets, we restricted ourselves to the *true* ACE mentions. Our focus is on evaluating the classification approaches versus the ranking approach rather than on building a full pronoun resolution system.

Following common practice in pronoun resolution, we report results in terms of *accuracy*, which is simply the ratio of correctly resolved anaphoric pronouns. Since the ACE data is annotated with coreference *chains*, we assumed that correctly resolving a pronoun amounts to selecting one of the previous elements in the chain as the antecedent.[20]

3.5.2 Comparative results

The results obtained for the four systems on the entire data set and the three ACE datasets are summarized in Table 3.3. They are compared with a naive baseline that picks the closest preceding mention as the antecedent.

System	ENTIRE ACE	BNEWS	NPAPER	NWIRE
Baseline	53.6	54.3	52.5	54.5
SCC$_1$	74.2	74.7	73.5	69.9
SCC$_2$	79.6	78.9	77.9	73.5
TCC	81.4	77.0	78.3	78.9
RK	82.4	80.3	79.2	79.5

Table 3.3: Accuracy scores for (Yang, 2005)'s single-candidate classifier (**SCC**$_1$), (Kehler et al., 2004a)'s single-candidate classifier (**SCC**$_2$), the twin-candidate classifier (**TCC**), and the ranker (**RK**).

As shown by this table, the ranker system outperforms the three classifier systems, with an accuracy of 82.4% on the entire ACE corpus. This corresponds to improvements

[20]This means that a pronoun can potentially be resolved to another pronoun.

of 8% and 2.8% (error reductions of 31% and 13%, respectively) over the single-candidate classifiers and of 1% (i.e., an error reduction of 5.4%) over the twin-candidate classifier.[21] But note that only the gains over the two single-candidate classifiers are statistically significant.[22] Although not significant, the gains over the twin-candidate classifier are however consistent across the different datasets.

3.5.3 Additional results

In this section, we discuss an additional experiment aimed at getting additional insight into the potential of the ranker. In the previous experiments, we provided a rather limited context for training: we only considered mentions in a window of 3 sentences around the correct antecedent. Our main motivation for doing this was to stay as close as possible to the testing conditions given in (Yang, 2005) for the twin-candidate approach, thereby giving it the fairest comparison possible. As noticed, this model is computationally much more intensive than the other models, making it difficult within this model to widen the candidate set during training and testing. The comparison of the two single-candidate models however suggests that extending the context lead to better performance: the model that uses the largest training and test windows outperforms the model that uses a smaller context. An open question is whether the ranker can also benefit from widening the window of candidates. To answer this question, we ran an experiment on the same three ACE datasets and widened the window of sentences by collecting, in addition to the closest antecedent, all non-antecedents preceding the anaphor up to 10 sentences before the antecedent (the test window was also widened accordingly). The results for this experiment are reported in table Table 3.4:

These figures show slight, although not significant, improvements on the entire ACE dataset and on the three datasets, with an overall score of 83.1%: the largest gain is found on the BNEWS where there is an error reduction of 8.1%. Note finally that using the train-

[21] Note that (Kehler et al., 2004a)'s original implementation had accuracy of 75.7% on the entire ACE data using a similar feature set. A difference however is that they didn't use the true ACE markables.

[22] Throughout our experiments, significance was examined by running a t-test, with $p < 0.05$.

System	ENTIRE ACE	BNEWS	NPAPER	NWIRE
RK ($w = 2$)	82.4	80.3	79.2	79.5
RK ($w = 10$)	83.1	81.9	80.1	80.1

Table 3.4: Accuracy scores for the ranker (RK) with a window of 10 sentences.

ing/test settings of $\mathbf{SCC}_1$ (that is, an even larger context) didn't yield additional improvements.

3.5.4 Learning curves

An important question is how the size of the training data impacts the performance of the various systems. Given the cost associated with the annotation of anaphora and coreference, this issue can be crucial in the choice of a system for a new language or a new domain. In order to address this question, we tested the different pronoun resolvers on the NPAPER dataset using different numbers of training documents.[23] Figure 3.1 plots the learning curves of the different systems.

The ranker outperforms all the other systems even when the number of training documents is as small as 20 documents. Beyond that point, the ranker systematically beat the other systems. The number of documents to outperform the **SCC** models is less than that, since the ranker consistently beat these models with 10 or more documents. Finally, note that the learning curves for the different models all show a tendency to plateau rather quickly (at 35 documents): this suggests that the current feature set is probably not rich enough.

3.6 Conclusions

We have demonstrated that using a ranking model for pronoun resolution performs better than a classification model. On the three ACE datasets, the ranker achieves error reduc-

[23] The NPAPER dataset is the largest among the three datasets, with $1,591$ third person pronouns for training and 457 for testing.

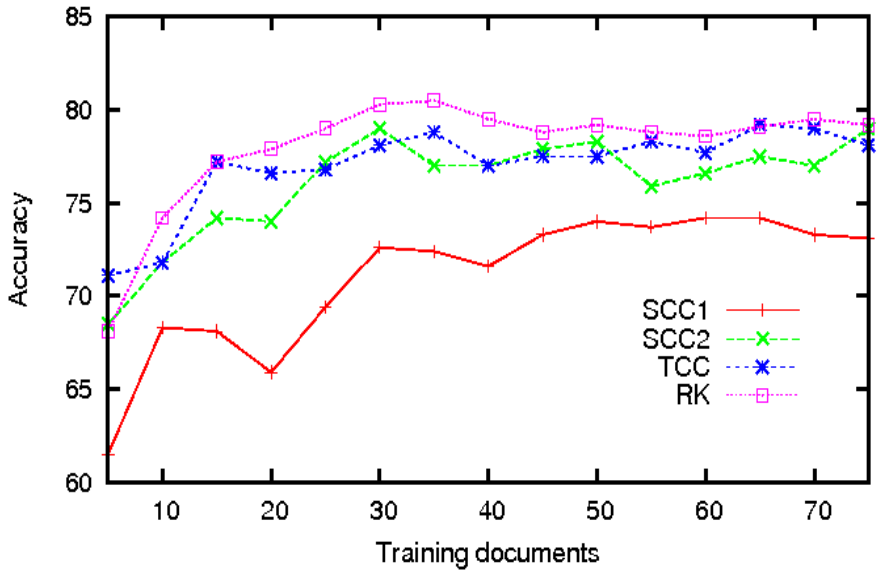

Figure 3.1: Learning curves of $\mathbf{SCC}_1$, $\mathbf{SCC}_2$, **TCC**, and **RK** for the NPAPER dataset.

tions ranging from of 31%, 13%, and 5.4% over the different classifier models. Our results thus corroborate Ravichandran et al.'s (Ravichandran et al., 2003) similar finding that ranking outperforms classification for question-answering. Clearly, the ability to consider all potential antecedents together, rather than independently, provides the ranker with greater discriminating power.

The main difference between the twin-candidate approach and the ranking approach is that under the former, candidates are compared by pairs (the best candidate is the one that has won the most times), whereas in the latter an ordering is imposed on the entire set at once. A potential advantage of the ranking approach is that it could allow one to define features on the candidate set itself. Another advantage of the ranker over the preference classifier is how ranking is obtained: only the ranker guarantees a global winner.

Besides performing better, the ranker is also the most attractive system from a strictly computational perspective. The round robin nature of the pairwise contests in the twin-candidate approach imposes a restrictive computational cost on its use which limits the

number of NP mentions that can be considered in a candidate set (both during training and testing). The ranker does not suffer from this limitation, and in fact we show that the ranker achieves a further error reduction of 8.1% by increasing the size of the candidate set used in training and testing. While the ranker has the same complexity as the single-candidate classifier, it is however slightly faster to train and test since the ranker uses only half the number of features used by the single-candidate classifier. Finally, we have shown through the use of learning curves that the ranker has a fast learning rate and does not require a lot of training data to outperform the classification-based models.

There are a number of ways that this model can be improved. First, notice that the feature set that was used is still rudimentary: in particular, it includes very little syntactic information (since it is here approximated in terms of POS contexts). Access to syntactic configurations and grammatical roles is likely to improve performance (see e.g., Yang et al. (2006)). While the ranker outperforms the classifiers outright, some benefit might also be gained by using both approaches together. It would be straightforward to integrate classifiers and rankers in an ensemble model. For example, a ranker could use the results of the classifier as features in its model.

Chapter 4

Extending the ranker to coreference resolution

This chapter extends the ranking approach proposed for pronoun resolution to the larger problem of coreference resolution. This extension consists in two important modifications, both motivated by the more complex nature of the coreference task. First, we create specialized ranking models for different classes of referential expressions, in particular: (i) third person pronouns, (ii) speech pronouns (i.e., first and second person pronouns), (iii) proper names, (iv) definite descriptions, (v) other types of phrases. Second, we augment these various rankers with a classifier model that predicts the discourse status of each mention. Specifically, this model is used as a filter for the different expert models: that is, all and only the mentions classified as discourse-old are resolved through their respective ranker. Evaluated on the ACE datasets, this simple cascade strategy yields significant improvements over a standard classifier-based coreference system on the three metrics described in Chapter 2.

4.1 Introduction

In the previous chapter, we have shown that a ranking model provides a theoretically more adequate and empirically better alternative approach to pronoun resolution than the traditionally used classification-based approaches. An open question is whether the superior antecedent selection capabilities offered by the ranker can also benefit in the larger task of coreference resolution. The nature of this task introduces two important challenges. The first extra difficulty introduced by the coreference task is that we are now dealing with various possible types of anaphoric expressions: in addition to third person pronouns, we now have to also handle speech pronouns (i.e., first and second person pronouns), proper names, definite descriptions, as well as other types of nominals (e.g., anaphoric uses of indefinite, quantified, and bare NPs). A large body of literature by theoretical linguists and psycholinguists suggest that different anaphoric expressions exhibit different patterns of resolution and are sensitive to different factors ((Ariel, 1988; Gundel et al., 1993) *inter alia*).[1] Most machine learning approaches have largely ignored these differences and have handled these different phenomena through a single monolithic model. A few exceptions are worth noting, though. Thus, (Morton, 2000) and (Ng, 2005a) propose different (classification) models for different NPs for coreference resolution and pronoun resolution, respectively. Other approaches (e.g., (Ng and Cardie, 2002a; Uryupina, 2004)) can be seen as partial attempts to capture the differential preferences between different anaphors by using different sample selection strategies during training. In this chapter, we propose different specialized ranker models corresponding to different types of referential expressions. In particular, we create models for: (i) third person pronouns, (ii) speech pronouns, and (iii) proper names, (iv) definite descriptions, and (v) all the others. These models are developed in Section 4.2.

The second challenge introduced by coreference resolution is that not all referen-

[1] Formal semanticists often distinguish pronouns, definite descriptions, and proper names in terms of their *presuppositional* behaviors: roughly, pronouns are most often bound, definite descriptions can be either bound or accommodated, and proper names are most often accommodated (e.g., van der Sandt (1992)). Note that the idea of treating definite descriptions and proper names as anaphors is fairly recent within formal semantics: the former were first treated as Russelian descriptions, while the later were treated as so-called *rigid designators*.

tial expressions in a given document are anaphors: some expressions introduce a discourse entity, rather than accessing an existing one. Thus the question of preventing the resolver to link these "discourse-new" expressions becomes an issue. Note that this is in principle a problem for any approach that tackles coreference resolution as a sequence of anaphora resolutions. This problem is easily handled in the standard classification approach (i.e., the single-candidate classifier): a mention will not be resolved if none of its candidates is classified positively. From this perspective, the pairwise classification model can be viewed as doing both discourse-status determination and resolution in a single step. The problem is however more troublesome for the ranker (or the twin-candidate classifier for that matter), which (by definition) always pick(s) out an antecedent. There are a number of possible scenarios to address this issue.[2] A natural solution is to use a model that specifically predicts the discourse status (discourse-new vs. discourse-old) of each expression: only the expressions that are classified as "discourse-old" by this model are considered by the rankers. Interestingly, this strategy has been used (unsuccessfully) as an attempt to improve the performance of the standard pairwise model ((Ng and Cardie, 2002b),(Ng, 2004)).[3] A variation of this approach would be to use a classifier model to output a list of valid candidates (i.e., those classified positively by the classifier) and use the ranker to identify the best among this list. This use of the ranker (a re-ranker in this case) is reminiscent of the work on parse selection which we mentioned in Chapter 3. Another solution is to use a threshold score: only the mentions for which at least one of the antecedent candidates meets a specified score are resolved. This option has the apparent advantage that no additional model needs to be created, but determining a proper threshold still requires additional experimentation. Furthermore, it is unclear whether a particular threshold will generalize well on new data. Maybe more problematic is that a threshold might not be able to distinguish between cases, where there is no good antecedent at all from cases in which the ranker is simply unsure

[2] See also (Yang, 2005) for a related discussion concerning the twin-candidate model.

[3] The detection of the discourse-new/discourse-old contrast has also generated research outside the context of coreference resolution: see for instance (Poesio et al., 2004).

about several, potentially good antecedents. Yet another solution is inspired by (Morton, 2000) who uses this option in the context of the pairwise classifier: it relies on the inclusion during training and testing of a "dummy" candidate, which serves as the antecedent for discourse-new expressions. In the rest of this chapter, we only discuss the first scenario (namely, the use of a discourse status determination module), leaving the others for future work. The discourse status determination module is presented in Section 4.3.

4.2 Learning specialized rankers

4.2.1 Linguistic motivations

In order to design different specialized models corresponding to different anaphoric expressions, one first has to decide along which dimension to split these expressions. A variety of options are in principle possible. As in (Ng, 2005a), one could for instance decide to learn a model for each set of anaphors that are lexically identical. That is, (Ng, 2005a) learns a model for *I*, *he*, *they*, and so on. While this option is possible for a closed category like pronouns, it is untenable in practice for other types of anaphors like proper names and definite descriptions. An important practical desideratum for acquiring adequate models is indeed to have sufficient data for training each of the models. Ideally, one would like to learn the classes of model that provide the best performance. Determining the optimal classes of models could potentially be achieved based on experimentation, but this is rather tedious and our models may not be able to generalize well. Instead, one could simply guide our split based on the particular *linguistic form* of the different expressions, as signaled for instance by the head word category and the determiner (if any).

That there is a correlation between the form of a referential expression and its anaphoric behavior is actually central to various linguistic and psycholinguistic theories ((Clark, 1975; Prince, 1981; Ariel, 1988; Gundel et al., 1993) *inter alia*). Basically, the idea is that linguistic form is an indicator of the status of the corresponding referent in the

discourse model. That is, the use by the speaker of a particular linguistic form corresponds to a particular level of activation (or familiarity or salience or accessibility) in (what she thinks is) the addressee's discourse model. For many authors, the relation takes the form of a continuum and is often represented in the form of a referential hierarchy. For instance, Ariel's "accessibility hierarchy" is given below:

> **Accessibility Hierarchy** (Ariel, 1988)
> Zero pronouns >> Pronouns >> Demonstrative pronouns >> Demonstrative NPs >> Short PNs >> Definite descriptions >> Full PNs >> Full PNs + appositive

The higher up, the more accessible (or salient or familiar), and the lower down the hierarchy, the less accessible (or salient or familiar) the entity. At the extremes of the hierarchy stand pronouns (these forms typically require a previous mention in the local context) and proper names (these forms are often used without previous mentions of the entity). This type of hierarchy is validated by corpus studies of the distribution of different types of expressions. For instance, (Ariel, 1988) who relies on recency as an estimation of salience (or accessibility in her terminology) shows that pronouns find their antecedents very locally (in a window of 1-2 sentences), while proper names predominantly find theirs at longer distances. Using discourse structure, (Asher et al., 2006) show that while anaphoric pronouns systematically obey the right-frontier constraint (i.e., their antecedents have to appear on the right edge of the discourse graph), this is less so for definites, and even less so for proper names.

From a machine learning perspective, these findings suggest that features encoding salience (e.g., distance, syntactic context) are likely to receive different sets of parameters depending on the form of the anaphor. This therefore suggests that better parameters are likely to be learned in the context of different models.[4] While the above studies focus primarily on salience, there are of course other dimensions according to which anaphors differ

[4]Another possible approach would consist in introducing different salience-based features encoding the form of the anaphor.

in their resolution preferences. Thus, the resolution of lexical expressions like definite descriptions and proper names is likely to benefit from the inclusion of features that compare the strings of the anaphor and the candidate antecedent (e.g., string matching) and features that identify particular syntactic configurations like appositive structures. This type of information is however much less likely to help in the resolution of pronominal forms. The problem is that, within a single model, such features are likely to receive strong parameters (due to the fact that they are good predictors for lexical anaphors) in a way that might eventually hurt pronominal resolutions.

In the following, we propose different ranking models corresponding to five types of referential expressions: (i) third person pronouns, (ii) speech pronouns, (iii) proper names, (iv) define descriptions, and (v) others (i.e., all expressions that don't fall into the previous categories). Note that this split only partially maps the referential hierarchy of (Ariel, 1988). Thus, there is no separate model for demonstrative NPs and pronominal forms: the main reason is that demonstrative NPs are very rare in the corpus we used (i.e., the ACE corpus).[5] These expressions were handled through the "others" model.[6] There is however a model for first and second person pronouns (i.e., speech pronouns): this is justified by the fact that these pronouns behave differently from their third person counterpart. These forms indeed often behave like deictics (i.e., they refer to discourse participants) or they appear within a quote.

[5]There are only 114 demonstrative NPs and 12 demonstrative pronouns in the entire ACE training.

[6]From a linguistic point of view, it would probably have made more sense to use one of the other models for these (e.g., the third person pronoun model for the demonstrative pronouns and the definite description model for the demonstrative NPs).

4.2.2 Ranking models

All the models are ranking models and they take the following generic log-linear form, repeated below for convenience from Chapter 3:

$$P_{rk}(\alpha_i|\pi) = \frac{\exp \sum_{j=1}^{m} w_j f_j(\pi, \alpha_i)}{\sum_k \exp \sum_{j=1}^{m} w_j f_j(\pi, \alpha_k)} \tag{4.1}$$

where π stands for the anaphoric expression, α_i for an antecedent candidate, f_j the weighted features of the model. The denominator consists of a normalization factor over the k mentions present in the candidate set. As before, model parameters were estimated with the limited memory variable metric algorithm implemented in TADM (Malouf, 2002). Gaussian smoothing was used to avoid extreme parameter values.

For the training of the different ranking models, we use a procedure similar to that described in Chapter 3. That is, for each model, instances are created by pairing each anaphor of the proper type (e.g., definite description) with a set of candidates which contains: (i) a true antecedent, and (ii) a set of non-antecedents. The selection of the true antecedent varies depending on the model we are training: for pronominal forms, the antecedent is selected as the *closest* preceding mention in the chain; for non-pronominal forms, we used the closest preceding *non-pronominal* mention in the chain as the antecedent.[7] For the creation of the non-antecedent set, we simply follow the approach in Chapter 3: in this set are collected all the non-antecedents that appear in a window of 2 sentences around the antecedent.[8]

[7] This sample selection has been proposed by (Ng and Cardie, 2002a) in the context of the standard approach. See discussion in Chapter 2.

[8] We suspect however that different sample selections might be more appropriate for different types of expressions.

4.2.3 Feature sets

This section describes the feature set used in the different ranking models. As in Chapter 3, our feature extraction relies on limited linguistic processing: we only made use of a sentence detector, a tokenizer, a POS tagger (as provided by the OpenNLP Toolkit[9]) and the Wordnet database[10]. Table 4.1 describes in detail the entire feature set, while Table 4.2 shows which features were used for which models.

First, we use the same five categories of features that were used for pronoun resolution, repeated below for convenience:

Linguistic form: This includes features pertaining to the referential form of the antecedent candidate: in particular, whether it is a proper name, a definite description, an indefinite NP, or a pronoun.

Context: This includes features describing the context of the antecedent candidate: these features can be seen as approximations of the grammatical roles, and as such inform us on the salience of the potential candidate (Grosz et al., 1995). For instance, we include as features the part of speech tags surrounding the candidate, as well as a feature that indicates whether the potential antecedent is the first mention in a sentence (approximating subject-hood), and a feature indicating whether the candidate is embedded inside another mention.

Distance: This includes features capturing the distance between the anaphor and the potential antecedent: pronouns due to their lack of lexical meaning are known to favor antecedents that are close-by (e.g., (Ariel, 1988; McEnery et al., 1997)). More specifically, we measured distance both in terms of the number of sentences and mentions intervening between them.

Morphosyntactic agreement: This includes features that encode the gender, number, and

[9] `opennlp.sf.net`.
[10] `http://wordnet.princeton.edu/`

Linguistic Form	
pn	α is a proper name {1,0}
def_np	α is a definite description {1,0}
indef_np	α is an indefinite description {1,0}
pro	α is a pronoun {1,0}
Context	
left_pos	POS of the token preceding α
right_pos	POS of the token following α
surr_pos	pair of POS for the tokens surrounding α
Distance	
s_dist	Binned values for sentence distance between π and α
np_dist	Binned values for mention distance between π and α
Morphosyntactic Agreement	
gender	pairs of attributes {masc, fem, neut, unk} for π and α
number	pairs of attributes {sg, pl} for π and α
person	pairs of attributes {1, 2, 3, 4, 5, 6} for π and α
Semantic compatibility	
wn_sense	pairs of Wordnet senses for π and α
String similarity	
str_match	π and α have identical strings {1,0}
left_substr	one mention is a left substring of the other {1,0}
right_substr	one mention is a right substring of the other {1,0}
hd_match	π and α have the same head word {1,0}
Apposition	
apposition	π and α are in an appositive structure {1,0}
Acronym	
acronym	π is an acronym of α or vice versa {1,0}

Table 4.1: Feature selection for the ranker models

person of the two mentions. These are determined for non-pronominal NPs using heuristics based on the part of speech tags (e.g., NN vs. NNS for number) and the actual strings of the mentions (e.g., whether the mention contains a male/female first name or honorific for gender). These features take the form of pairs of attributes, making sure that not only strict agreement (e.g., *singular-singular*) but also mere

compatibility (e.g., *masculine-unknown*) is captured.

Semantic compatibility: This includes features designed to assess whether the two mentions are semantically compatible. For these features, we use the Wordnet database: in particular, we collected the synonym set (or synset) as well as the synset of their direct hypernyms associated with each mention. In the case of common nouns, we used the synset associated with the first sense associated with the mention's head word. In the case of proper names, we used the synset associated with the name if available, and the string itself otherwise. For pronouns (which are not part of Wordnet), we simply used the pronominal form.

All these features were used in all five models. While one may question the use of distance for non-pronominal anaphors,[11] we think that their inclusion is justified by the fact that they might predict some "obviation" effects. As claimed by (Ariel, 1988) and others, definite descriptions and proper names are sensitive to distance too, although not in the same way as pronouns are: they show a preference for antecedents that appear outside a window of 1 or 2 sentences.

In addition to these core features, we add several other features which are only used by specific models (in particular, the models for definite descriptions and proper names):

String similarity: This includes features that test how similar the anaphor's and the antecedent candidate's strings are. Examples are perfect string matching (i.e., the two mentions are identical), substring matchings (i.e., one of the mentions is a substring of the other), and head matching (i.e., the two mentions share the same head word). These features are only used in the three non-pronominal models.

Appositive: This feature tests whether the anaphor is an apposition of the antecedent candidate. Since we don't have access to syntactic structure, we used various heuristics

[11] In fact, (Morton, 2000) doesn't use distance features in this case.

(e.g., the presence of a comma between the two mentions) to compute this feature. This feature was used only by the proper name and definite NP models.

Acronym: This feature determines whether the anaphor's string is an acronym of the antecedent candidate's string (and vice versa): e.g., NSA and National Security Agency. This feature was used only by the proper name model.

Features/Types	3^{rd} pron.	speech pron.	proper names	def. NPs	others
Ling. form	√	√	√	√	√
Context	√	√	√	√	√
Distance	√	√	√	√	√
Morphosynt. agr.	√	√	√	√	√
Sem. compat.	√	√	√	√	√
Str. sim.			√	√	√
Apposition			√	√	
Acronym			√		

Table 4.2: Features used in modeling each class of referential expressions

4.2.4 Antecedent selection results

In this section, we report the performance of the different ranker models with respect to anaphora resolution. That is, we specifically evaluate the ability of each resolver of selecting a correct antecedent for each anaphor. The training and testing datasets used for our experiments come from the ACE corpus, as described in Chapter 2. The total number of anaphors (i.e., of mentions that are not chain heads) in the data is $19,322$ and $4,599$ for training and testing, respectively. The distribution of each anaphoric type is presented in Table 4.3. Roughly, third person pronouns account for 22-24% of all anaphors in the entire corpus, speech pronouns for 11-13%, proper names for 33-40%, and definite descriptions for 16-17%. The distribution is slightly different from one dataset to another, probably reflecting genre differences. For instance, BNEWS shows a larger proportion of pronouns in general (pronominal forms account for 40-44% of all the anaphoric forms).

Type/Count	ENTIRE ACE		BNEWS		NPAPER		NWIRE	
	train	test	train	test	train	test	train	test
3^{rd} pron.	4,389	1,093	1,419	304	1,591	457	1,379	332
speech pron.	2,178	610	1,056	330	373	158	749	122
proper names	7,868	1,532	1,902	448	3,386	534	2,580	550
def. NPs	3,124	796	858	250	1,155	271	1,111	275
others	1,763	568	361	225	716	230	686	203
Total	19,322	4,599	5,596	1,557	7,221	1,560	6,505	1,482

Table 4.3: Distribution of the different anaphors in ACE

For this set of experiments, we used exactly the same development cycle as described in Chapter 3. In testing the different systems, we again assume perfect mention boundaries: only the true ACE mentions were considered as potential candidates. The candidate set during testing was formed by taking *all* the mentions that appear before the anaphor. Also, we assumed that correctly resolving an anaphor amounts to selecting one of the previous mentions in the entity as the antecedent. The accuracy scores for the different models are presented in Table 4.4.

System	ENTIRE ACE	BNEWS	NPAPER	NWIRE
3^{rd} pron.	82.2	81.6	80.4	80.2
speech pron.	66.9	64.8	63.9	58.2
proper names	83.5	81.5	81.5	85.3
def. NPs	66.5	67.6	67.9	57.5
others	63.6	62.1	57.0	62.8

Table 4.4: Accuracy of the different ranker models

The best accuracy results on the entire ACE corpus are found first for the proper name resolver with a score of 83.5%, then for the third person pronoun resolver with 82.2%, then for the definite description and speech pronoun resolvers with 66.9 and 66.5 respectively. The worst scores are obtained for the "others" category. This pattern is not really surprising. The high scores for the third person pronoun and the proper name rankers most likely follow from the fact that the resolution of these expressions relies on "cheap" and reliable predictors, such as distance and morphosyntactic agreement for pronouns, and string

similarity features for proper names. The resolution of definite descriptions and other types of lexical NPs (which are handled through the backup "others" model) are much more challenging in relying on lexical semantic and world knowledge information, which is only partially encoded via our Wordnet-based features. Finally, note that the resolution of speech pronouns is also much harder than that of the other pronomimal forms: these expressions are much less (if at all) constrained by recency and agreement. Furthermore, these expressions show a lot of cataphoric uses (e.g., in structures like *"My energy policy encourages conservation," declared George Bush*), which are not considered by our models. The low scores for the "others" category is attributable to the fact that this model, which works as a sort of backoff model, encompasses very different referential expressions.

4.3 Predicting discourse status

We now turn to the presentation of the model used for determining the discourse status of mentions, starting with the form of the model and then describing the feature selection.

4.3.1 Classification model

The task for the discourse status determination component is the following: one wants to decide for each mention α in a document whether α is discourse-new (i.e., the mention introduces a new entity) or discourse-old (i.e., the mention accesses an existing entity). This task can be performed using a simple classifier with two possible outputs: NEW and OLD. The classifier estimates the conditional probabilities $P(c|\alpha)$, where $c \in \{\text{NEW}, \text{OLD}\}$, and predicts the class that receives the highest score. This model takes the following log-linear form:

$$P_{ds}(\text{OLD}|\alpha) = \frac{\exp \sum_{j=1}^{m} \lambda_j f_j(\alpha, \text{OLD})}{\sum_{c \in \{\text{NEW},\text{OLD}\}} \exp \sum_{j=1}^{m} w_j f_j(\alpha, c)} \tag{4.2}$$

where $f_j(\alpha, c)$ is the number of times feature j occurs for mention α, and w_j is the weight assigned to j during training. The denominator consists of a normalization factor over the two possible outcomes NEW and OLD. Model parameters are estimated with the limited memory variable metric algorithm implemented in TADM (Malouf, 2002). Gaussian smoothing was used to avoid extreme parameter values. The training procedure for creating this model is very straightforward: the set of mentions $\mathcal{M}$ in each document is iterated over and each mention α is assigned a label: NEW if α is the head of a chain (this includes single-mention entity) or OLD otherwise.

4.3.2 Feature set

For constructing our discourse status classifier, we rely on three main types of information sources. Our feature set is similar, although not identical, to that proposed by (Ng and Cardie, 2002a). First, we design features that describe the mention itself, ranging from the number of tokens in the mention to finer-grained features encoding the linguistic form of the mention. The first set of features are directly inspired by "accessibility hierarchy" above: there is indeed a correlation between both the lexical "heaviness" and the form of an expression and its discourse status. For instance, shorter expressions are more likely to access entities that are already in the discourse model (i.e., to be discourse-old). A second set of features pertains to the position of the mention in the text: in particular, we rely on the intuition that expressions mentioned earlier are more likely to be discourse-new. Finally, a third set of features compares the given mention to the mentions that precede it in the text. Examples include whether or not the mention's string matches that of a preceding mention, and whether or not the mention appears in particular configuration like appositive structures.

Word count	
wd_count	number of words in α {1,2,3,...}
Linguistic form	
pro	α is a pronoun {1,0}
speech_pro	α is a speech pronoun {1,0}
refl_pro	α is a reflexive pronoun {1,0}
pn	α is a proper name {1,0}
short_pn	α is a single word proper name {1,0}
def_np	α is a definite description {1,0}
short_def_np	α is a single noun definite description {1,0}
indef_np	α is an indefinite description {1,0}
quant_np	α is a quantified description {1,0}
poss_np	α is a possessive description {1,0}
bare_np	α is a bare noun {1,0}
rel_cl	α contains a relative pronoun {1,0}
Position in text	
first_sent	α appears in the first sentence {1,0}
first_5_sent	α appears in the first 5 sentences {1,0}
first_10_sent	α appears in the first 10 sentences {1,0}
Relation to previous mentions	
embedding	α is embedded within another mention {1,0}
str_match	α's string matches that of a previous mention {1,0}
hd_match	α's head word matches that of a previous mention {1,0}
apposition	α's is an apposition to a previous mention {1,0}
acronym	α's is an acronym of a previous mention (or vice versa){1,0}

Table 4.5: Feature selection for the discourse status model

4.3.3 Results

In this section, we report on the performance of the discourse status classifier. This system was evaluated on the ACE datasets, training the model on the `train` texts, and applying the classifier to the `devtest` texts. As a baseline measurement, we used the majority class (OLD in this case); this strategy obtains an accuracy of 59.7% on the entire ACE corpus. The discourse status model, on the other hand, achieves an overall accuracy score of 80.8%. The results for the model trained/tested on the different datasets are as follows: 80.1 for BNEWS,

82.2 NPAPER, and 81.1 for NWIRE.[12]

It is instructive to look at the errors made by the discourse status determination classifier. Error analysis on the development data reveal that the model successfully identifies 78.8% of the true anaphors, and 83.8% of the non-anaphors. That is, the classifier does a better job at detecting discourse-new entities than discourse-old ones. In terms of errors made for the different types of expressions, we found that the most errors were made in classifying non-pronominal forms: in particular, proper names and definite descriptions. Thus, misclassified proper names account for 35.3% of the "missed anaphors" (i.e., anaphors misclassified as NEW) and 29.5% of the "spurious anaphors" (i.e., non-anaphors misclassified as OLD). Misclassified definites account for 22.8% and 29.5% of these errors, respectively. This makes a certain amount of sense given that these expressions are more versatile than pronouns in terms of their discourse status. While a vast majority of pronouns are anaphoric (95.7% for third person pronouns, and 79.4 for speech pronouns), definite descriptions and proper names are often ambiguous in terms of their discourse status (57.8% of definites and 58.7% of proper names are anaphoric). Note finally that quite of few speech pronouns were also misclassified in being wrongly identified as NEW: they account for 18.5% of the "spurious anaphors".

4.4 Experiments

4.4.1 System architecture

We are now ready to deploy the ranking approach to the task of coreference resolution. The overall system architecture is straightforward. For each mention $m \in \mathcal{M}$ encountered in the current document D, the discourse status model is first applied to determine whether m introduces a new discourse entity (i.e., it is classified as NEW) or refers back to an existing entity (i.e., it is classified as OLD). If m is classified as NEW, the process terminates and

[12]These results are slightly below the results of (Ng and Cardie, 2002a), who report score of $\sim 85\%$ on the MUC-6 and MUC-7 datasets.

goes to the next mention. If m is classified as OLD, m along with its set of antecedent candidates $\mathcal{C}_m$ is sent to the corresponding resolver (e.g., the third person pronoun model if m is a third person pronoun) which picks the "best" candidate among $\mathcal{C}_m$. The candidate set here includes *all* the mentions that linearly precede m. The output of the system consists of a list of mention pairs (i.e., the coreference links) which in turn defines (through reflexive, transitive closure) a partition over the set of mentions $\mathcal{M}$ in D. In the following, we will refer to this coreference system as **ERK+DS**.

4.4.2 Baseline systems

In this section, we present four baseline coreference systems against which we will evaluate **ERK+DS**. All these systems are variations on the standard approach described in Chapter 2, and are based on the pairwise classification approach: that is, they are all single-candidate classifiers.

SCC This first system is an implementation of the standard approach described in Chapter 2. In particular, we follow the training and test procedures proposed by (Ng and Cardie, 2002a). During training, instances are formed by pairing each anaphor with each of its preceding candidates, until the antecedent is reached: the closest preceding antecedent in the case of a pronominal anaphor, or the closest non-pronominal antecedent in the case of a non-pronomninal anaphor. During testing, instances are formed by pairing each mention with each of its preceding mentions. Each instance is then submitted to the classifier, which determines whether the pair under inspection is coreferential or not. If none of the pairs created for a given mention is classified positively, the mention is left unresolved. If several pairs for a given mention are classified positively, then the pair with the highest score is selected (i.e., this is the "Best-First" link selection).

SCC+DS This second system augments the previous system with the discourse status classifier. That is, like **ERK+DS**, the discourse status model is first used to filter the

non-anaphors. In turn, all the mentions that are classified as anaphoric are sent to the coreference model, which is then used to produce an antecedent for each anaphor (i.e., the candidate with the highest score with respect to the COREF class). This system is very similar to the approach proposed in (Ng and Cardie, 2002b).[13]

ESCC This third system implements various single-candidate classifiers for the different referential types. That is, we built expert classification models for: (i) third person pronouns, (ii) speech pronouns, (iii) proper names, (iv) definite descriptions, (v) other types of phrases. The training and test procedures are the same as for the **SCC**.

ESCC+DS Finally, this last system augments the **ESCC** with the discourse status classifier. That is, the application of a given expert model to a given mention is conditioned on that mention being classified as OLD by the discourse status model.

The feature set used in the baseline systems includes all the features that were used for the rankers (Table 4.5). In accordance with how previous approaches have designed feature sets in the standard pairwise approach, we have also added extra features describing the linguistic form of the potential anaphor (whether it is a pronoun, a proper name, and so on). For the baseline systems that use expert models, that is **ESCC** and **ESCC+DS**, we use the same feature split as for the expert rankers (as described in Table 4.2).

4.4.3 Main Results

This section describes the performance of the **ERK+DS** in comparison to the different classifier-based systems. The different systems were trained and tested on the ACE corpus; we again assume perfect mention boundaries: only the true ACE mentions were considered

[13] An important difference is however that the system proposed in (Ng and Cardie, 2002b) does not necessarily yield an antecedent for each of the anaphors proposed by the discourse status model. In (Ng and Cardie, 2002b), the coreference classifier is applied as in **SCC**, which means some of the proposed anaphors might not be resolved (i.e., in the case where none of the pairs for that anaphor is classified positively). In this case, the coreference model can act as an additional filter. Not suprisingly, these authors report gains in precision but comparatively larger losses in recall. Our development experiments revealed that the approach implemented in **SCC+DS** provided a better performing baseline.

both during training and testing. For evaluating these systems, we use the three different coreference resolution metrics described in Chapter 2, namely: the MUC metric of (Vilain et al., 1995), the B^3 metric of (Bagga and Baldwin, 1998), and the CEAF metric of (Luo, 2005). The results for the entire ACE corpus are summarized in Table 4.6.

System	MUC			B^3			CEAF		
	R	P	F	R	P	F	R	P	F
SCC	60.8	72.6	66.2	62.4	77.7	69.2	62.3	62.3	62.3
SCC+DS	64.9	72.3	68.4	65.6	74.1	69.6	63.4	63.4	63.4
ESCC	64.8	74.5	69.3	65.3	79.1	71.5	65.0	65.0	65.0
ESCC+DS	66.8	74.4	70.4	66.4	77.0	71.3	65.3	65.3	65.3
ERK+DS	67.9	75.7	71.6	66.8	79.8	72.7	67.0	67.0	67.0

Table 4.6: Recall (R), Precision (P), and f-score (F) results on the entire ACE corpus using the MUC, B^3, and CEAF metrics

The first thing to note about these results is that the **ERK+DS** system significantly outperforms the different classifier-based systems on the three different metrics.[14] The f-scores for this system are 71.6% with the MUC metric, 72.7% with the B^3, and 67% with the CEAF metric. These scores place the **ERK+DS** among the best coreference resolution systems, since most existing systems are typically well under the bar of the 70% in f-score with the MUC and B^3metrics (Ng, 2005b). The fact that improvements are consistent across the different evaluation metrics is remarkable, especially given that these three metrics are quite different in the way they compute their scores. The gains in f-score range from 1.2 to 5.4% on the MUC metric (i.e., error reductions of 4 to 15.9%), from 1.4 to 3.5% on the B^3 metric (i.e., error reductions of 4.8 to 11.4%), and from 1.7 to 4.7% on the CEAF metric (i.e., error reductions of 6.9 to 17%).

The larger improvements come from recall, with improvements ranging from 1.9 to 7.1% with MUC, from 2.4 to 5.6% with B^3.[15] This suggests that **ERK+DS** is predicting a lot

[14] Statistical significance was examined by running a t-test for both recall and precision scores, with $p < 0.05$.

[15] Recall that recall and precision scores are identical with CEAF, due to the fact that we are using true mention boundaries. See Chapter 2 for details.

more valid coreference links than the baseline systems. Although smaller, significant gains are also made in precision: this means that **ERK+DS** is at the same time able to reduce the proportion of invalid links that are being produced. Both these improvements result in an overall better partition of the set of mentions.

These overall improvements found with the **ERK+DS** system can be attributed to the combination of two main factors. First, these results suggest that this system is able to capitalize on the better antecedent selection capabilities offered by the ranking approach. This is supported by the error analysis we performed on the development data. Errors made by a coreference system can be conceptualized as falling into three main classes: (i) "missed anaphors" (i.e., an anaphoric mention that fails to be linked to a previous mention), (ii) "spurious anaphors" (i.e., an non-anaphoric mention that is linked to a previous mention), and (iii) "invalid resolutions" (i.e., a true anaphor that is linked to a incorrect antecedent). The two first types of errors pertain to the determination of the discourse status of the mention, while the third type of errors pertains to the selection of an antecedent (i.e., anaphora resolution). When looking at the invalid resolutions made by the different systems, we found that the **ERK+DS** had a much lower error rate: only 17.9% of all true anaphors were incorrectly resolved by this system, against 23.1% for **SCC**, 24.9% for **SCC+DS**, 20.4% for **ESCC**, and 22.1% for **ESCC+DS**. Large error reductions were made, in order of magnitude, in the resolution of third person pronouns, definite descriptions and proper names. Interestingly, no error reduction was found in the resolution of speech pronouns.

The second factor responsible for the good performance of **ERK+DS** is in the use of specialized models. Having a separate, expert model for each type of referring expressions allows the various features to be weighted differently depending on the type of anaphors we are dealing with. This in turn provides the specialized models with additional discriminative power over a single model. The advantage of having specialized models can actually be seen by comparing the baseline systems that use separate models (**ESCC** and **ESCC+DS**) against those that use a single model (**SCC** and **SCC+DS**): in both cases, the system that

uses specialized models outperforms the counterpart system that relies on a single model with significant gains in both recall and precision.

The results on the different ACE datasets are given in Table 4.8-Table 4.10. Overall, these results show the same pattern as on the entire ACE corpus: the **ERK+DS** system consistently outperforms the baseline systems. The largest gains are made on NPAPER, while the smallest ones are made on BNEWS. We attribute the relatively poorer results on this later dataset to the high proportion of speech pronouns therein. As noted, the antecedent selection accuracy for these expressions was rather low and didn't improve from the use of a specialized ranker. In Figure 4.1, we report the different B^3 recall and precision rates for the different systems on the entire and the different ACE datasets.

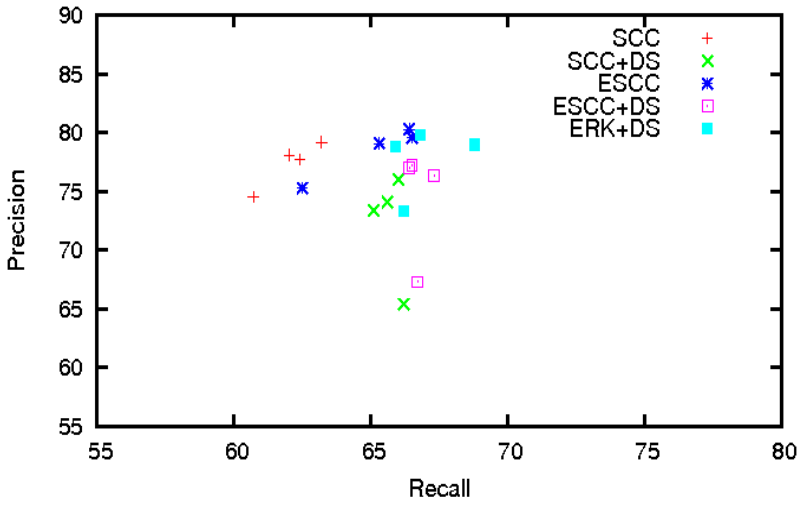

Figure 4.1: B^3 recall and precision of **SCC**, **SCC+DS**, **ESCC**, **ESCC+DS**, and **ERK+DS** on the entire and the three ACE datasets

Finally, it is instructive to compare the different baselines together. Two main patterns emerge from the comparison of these systems. First, as noted, the systems that use specialized coreference models (i.e., **ESCC** and **ESCC+DS**) respectively outperform the sys-

tems that rely on a single model (i.e., **SCC** and **SCC+DS**). The improvements are made both in recall and precision. Second, the systems that use a discourse status model (i.e., **SCC+DS** and **ESCC+DS**) also tend to produce f-score improvements over the models that don't (i.e., **SCC** and **ESCC**). This is at least true for two of the metrics: MUC and CEAF. Note that the gains there are exclusively made in recall, sometimes with important losses in precision (especially with B^3). The boost in recall suggests that the discourse status model has a positive effect in "rescuing" some true anaphors for which the coreference model(s) alone wouldn't have produced any coreference link. The drop in precision, on the other hand, suggests that not all these rescued anaphors are properly resolved by the classifier model(s). The fact that the precision losses are more important in B^3 than in MUC comes from the way these two metrics work. Recall than with B^3, errors are computed at the level of each mention: this means that the addition of invalid links to a chain will be compounded for each mention.

4.4.4 Oracle results

So far, we have shown that an approach combining the use of specialized rankers with a discourse status classifier yields coreference performance superior to those given by various classification-based baseline systems. Crucially, these improvements have been possible using a discourse status model that has an accuracy of just 80.8% (when trained and tested on the entire ACE data). Clearly, the performance of the discourse status module has a direct impact on the performance of the entire coreference system. On the one hand, misclassified anaphors are simply not resolved by the rankers: this limits the recall of the coreference system. On the other hand, misclassified non-anaphors are linked to a previous mention: this limits the precision of the coreference system.

In order to better assess the negative impact of the errors made by the discourse status classifier, we build two different oracle systems. The first oracle system, **ERK+DS-ORACLE**, uses the specialized rankers in combination with a perfect discourse status classifier. That is, this system knows for each mention whether it is anaphoric or not: in turn,

System	MUC			B^3			CEAF		
	R	P	F	R	P	F	R	P	F
ERK+DS	67.9	75.7	71.6	66.8	79.8	72.7	67.0	67.0	67.0
ERK+DS-ORACLE	79.1	79.1	79.1	75.4	76.0	75.7	76.9	76.9	76.9
LINK-ORACLE	78.8	100.0	88.1	74.3	100.0	85.2	79.7	79.7	79.7

Table 4.7: Recall (R), Precision (P), and f-score (F) results for **ERK+DS-ORACLE** and **LINK-ORACLE** on the entire ACE corpus

the only errors made by such a system are "invalid resolutions". From this perspective, **ERK+DS-ORACLE** provides us with an upper-bound for the **ERK+DS** approach. The results for this oracle are given in Table 4.7: they show substantial improvements over **ERK+DS**, which suggests that the **ERK+DS** has also the potential to be further improved if used in combination with a more accurate discourse status classifier.

The second oracle system, **LINK-ORACLE**, uses the discourse status classifier presented in section Section 4.3 with a perfect anaphora resolver. That is, this system has perfect knowledge regarding the antecedents of anaphors: the errors made by such a system are only errors in the discourse status of mentions. The results for **LINK-ORACLE** are also reported in Table 4.7. What these results mean is that however accurate our rankers get at picking a correct antecedent for a true anaphor, the best our system can achieve in terms of f-scores is: 88.1% with MUC, 85.2% with B^3, and 79.7% with CEAF.

4.5 Summary and discussion

In this chapter, we have proposed an extension of the ranking approach presented in Chapter 3 for pronoun resolution to the larger problem of coreference resolution. Relying on linguistic motivations, this extension consists in: (i) the creation of separate, expert ranker models corresponding to different types of referring expressions, and (ii) the use of discourse status classifier which determines the mentions that are sent to the rankers. This simple pipeline architecture results in significant improvements over various implementations of the standard, classifier-based coreference system. Importantly, these improvements

System	MUC			B^3			CEAF		
	R	P	F	R	P	F	R	P	F
SCC	60.9	73.2	66.5	63.2	79.2	70.3	63.8	63.8	63.8
SCC+DS	64.2	73.7	68.7	66.0	76.0	70.6	64.1	64.1	64.1
ESCC	64.9	76.9	70.4	66.4	80.3	72.7	66.4	66.4	66.4
ESCC+DS	65.6	75.3	70.1	66.5	77.2	71.4	64.9	64.9	64.9
ERK+DS	65.7	75.4	70.2	65.9	78.8	71.8	65.7	65.7	65.7

Table 4.8: Recall (R), Precision (P), and f-score (F) results on the BNEWS dataset using the MUC, B^3, and CEAF metrics

System	MUC			B^3			CEAF		
	R	P	F	R	P	F	R	P	F
SCC	63.0	72.9	67.6	60.7	74.5	66.9	59.9	59.9	59.9
SCC+DS	68.6	71.3	69.9	66.2	65.4	65.8	59.6	59.6	59.6
ESCC	64.5	73.5	68.7	62.5	75.3	68.3	61.5	61.5	61.5
ESCC+DS	69.4	72.2	70.8	66.7	67.3	67.0	61.0	61.0	61.0
ERK+DS	70.8	73.6	72.2	66.2	73.3	69.5	65.3	65.3	65.3

Table 4.9: Recall (R), Precision (P), and f-score (F) results on the NPAPER dataset using the MUC, B^3, and CEAF metrics

System	MUC			B^3			CEAF		
	R	P	F	R	P	F	R	P	F
SCC	58.2	69.3	63.2	62.0	78.1	69.1	61.9	61.9	61.9
SCC+DS	64.5	69.6	66.9	65.1	73.4	69.0	62.4	62.4	62.4
ESCC	64.1	72.7	68.1	66.5	79.6	72.5	66.4	66.4	66.4
ESCC+DS	66.6	71.8	69.1	67.3	76.4	71.6	65.7	65.7	65.7
ERK+DS	68.1	73.4	70.7	68.8	79.0	73.6	68.1	68.1	68.1

Table 4.10: Recall (R), Precision (P), and f-score (F) results on the NWIRE dataset using the MUC, B^3, and CEAF metrics

are consistent across the three main coreference evaluation metrics: MUC, B^3, and CEAF. We attribute the good performance of the proposed approach to: (i) the better antecedent selection capabilities offered by the ranking approach, and (ii) the division of labor between specialized models for different types of anaphors.

There are a number of ways to improve on the current approach. As seen in the oracle experiments, there is still a lot of room for improvement both on the side of the rankers and on the side of the discourse status classifier. The different ranking models can probably be enhanced both in terms of feature selection and in terms of the sampling of the training instances. For instance, we have noted that the ranker for speech pronouns didn't produce very high resolution scores. We suspect that this is due to the lack of adequate features for this type of expression. The use of additional semantic information, which can mined from web-based resources like Wikipedia (e.g., (Ponzetto and Strube, 2006)), is also likely to improve in the resolution of definite descriptions. As for training, we have used the same sample selection for the different types of expressions (modulo the split between pronominal and non-pronominal forms): using different sample selections is also likely to improve the performance of the different models (Uryupina, 2004). Finally, note that the split we used in building the different rankers is also likely not to be optimal. In particular, the "others" model covers very different types of expressions (from demonstratives pronouns and NPs to bare nouns to indefinite descriptions) which are likely to be better handled by different models. Some improvements are also possible on the side of the discourse status model. For instance, it would probably make sense to design different discourse status models for different types of referring expressions.

Despite its good performance, the approach proposed in this chapter only departs from the standard approach presented in Chapter 2 in the use of a different type of model: it uses a ranking function instead of a classification function. That is, the general approach still relies on the simplistic assumption that coreference resolution can be reduced to a sequence of anaphora resolutions. Under this view, the creation of the coreference chains is simply achieved through reflexive, transitive closure over the set of anaphor-antecedent pairs (where an anaphor is given exactly one antecedent). Notwithstanding its intuitive appeal, this method of clustering mentions is *ad hoc* and as such unlikely to be optimal in providing us with the best overall partition. First, note that this way of linking mentions

is very conservative: since only one antecedent is posited for each anaphor, the number of generated links is bound to be small. This explains why the results for such systems usually show high precision but comparatively much poorer recall. An important part of the problem with this approach is that it fails to ensure any sort of global coherence on the creation of the coreference chains. Resolutions are always made *independently* from one another: this potentially calls for situations in which, because of transitive closure, two incompatible mentions "accidentally" end up in the same chain. Finally, the interaction of the models in the pipeline architecture is also likely to be sub-optimal. As noted, a lot of true anaphors (over 21%) are left unresolved, while a lot of true non-anaphors (over 16%) are incorrectly forced to be resolved. The main problem here is that the decisions of discourse status model are always taken on faith by the rankers, irrespective of the internal confidence of the models. That is, a mention that is —maybe incorrectly— classified as anaphoric by the discourse status model is forced to be resolved, irrespective of the confidence the coreference model has with respect to its resolution. Similarly, a mention that is —maybe incorrectly— classified as non-anaphoric by the discourse status model is left resolved, irrespective of the confidence the coreference model might have with respect to its resolution. The problem is again that of making too strong independence assumptions, but this time between the discourse status model decisions and the coreference model decisions. Ideally, one would instead like the discourse status and the coreference models to mutually inform each other and make a common decision. We turn to these different issues in the next chapter.

Chapter 5

Coreference resolution as linear optimization

In this chapter, we show how the task of coreference resolution can be recast as a linear optimization problem. In particular, we use the framework of Integer Linear Programming (ILP) to: (i) combine the predictions of three local models (namely, a standard pairwise coreference classifier, a discourse status classifier, and a named entity classifier) in a joint, global inference, and (ii) integrate various other global constraints (such as transitivity constraints) to better capture the dependencies between coreference decisions. Tested on the ACE datasets, our ILP formulations deliver significant f-score improvements over both a standard pairwise model and various models that employ the discourse status and a named entity classifiers in a cascade. Improvements were found across the three different evaluation metrics: MUC, B^3, and CEAF. [1]

[1] This chapter is based on and extends (Denis and Baldridge, 2007b).

5.1 Introduction

The previous chapters have primarily focused on investigating the use of a different type of model (namely, ranking models) in order to improve the anaphora and coreference resolution. In particular, we have shown that the objective function used in ranking provides a more adequate way to model the process of antecedent selection, resulting in performance improvements in both tasks. The present chapter turns to two other problems that currently limit the performance of state-of-the-art coreference resolution systems.

The first problem is that of **knowledge prediction and integration**. As noted in Chapter 1, reference resolution depends on a multitude of information sources. Although machine learning systems have been reasonably successful by simply utilizing a few shallow features, it is generally agreed that drastic improvements will only be possible by incorporating a wider set of information sources (in particular, semantic and pragmatic ones). Quite a few approaches have actually tried to incorporate richer feature sets into their coreference system, but their results have been overall disappointing, sometimes leading to small improvements (Ponzetto and Strube, 2006; Yang et al., 2006; Ng, 2007), but also to degradation (Kehler et al., 2004a; Ng and Cardie, 2002b; Denis and Kuhn, 2006) in performance. The main problem faced by these approaches is that predicting linguistically rich information from raw text is challenging, which in turn means that their automatic extraction is likely to be noisy. This raises the question of how to best incorporate this imperfect information into our coreference system.

In this chapter, we propose to enrich a standard coreference model with information coming from two main information sources: discourse status information and name entity information. These are predicted through separately learned models. Intuitively, we only should identify antecedents for the mentions which are likely to have one (i.e., discourse-old mentions) (Ng and Cardie, 2002b), and we should only make a set of mentions coreferent if they all have the *same* entity type (eg, PERSON or LOCATION). Richer information of this sort has generally been incorporated into coreference systems either as pre- or post-

processing modules during search or in the form of features in the coreference model. Both of these approaches are problematic, as noted in Chapter 1. They fail to model the complex dependencies between the different models; this leads to a situation in which one model (and the errors it makes) over-constrains the other.

The use of a discourse status classifier in a cascade with different coreference models (i.e., rankers and classifiers) was discussed in the previous chapter. As pointed out, augmenting the coreference classifier(s) with a discourse status filter provides only small (if any) f-score improvements due to the fact that: (i) many true anaphors were left unresolved, and (ii) many true non-anaphors were resolved. Interestingly, Ng (2004) reports that incorporating discourse status information in the form of (binary) features also fails to provide decisive improvements. This author in turn proposes to tune the classification threshold used by the discourse status model in a way that provides improvements on the coreference task.[2] While it achieves global optimization over the two models, this method involves a fair amount of tuning.

In the following, we suggest a different approach for combining the predictions of the various classifiers. That is, we treat the three tasks of discourse status determination, named entity classification, and coreference resolution as a *joint* problem. Specifically, the outcomes of the three locally learned models are represented as a collection of random variables for which we seek an optimal global assignment. This optimization is subject to a set of declarative constraints that encode the dependencies between the models and that have the effect of mutually constraining their final outcomes. We use Integer Linear Programming (ILP) to cleanly integrate the predictions of the local models and to perform the global inference over these models. A crucial advantage of the ILP approach over that of Ng (2004) is that it does not require careful weighting of the models (though this can be done) —the emphasis is instead on ensuring *consistency* between model assignments.

The second problem that we address in this chapter is that of **locality of the corefer-**

[2] In Ng (2004), the best probability threshold for the class OLD is .3 (instead of .5). Concretely, this means that some additional "anaphors" are submitted to the coreference model, leading to recall increases.

ence decisions. In the standard coreference classification approach (or the ranking approach for that matter), both the classification and clustering decisions are made solely based on pairs of mentions: that is, the coreference decisions are made independently of one another. This is clearly a simplification. The different coreference decisions should instead be conditioned on how well it matches the entity as a whole (McCallum and Wellner, 2003). This problem has motivated different authors to explore globally trained models, in which coreference decisions are conditioned on entities (i.e., chains), rather than on mentions (Morton, 2000; Luo et al., 2004; Culotta et al., 2007). This has the advantage of allowing one to define larger features, and therefore ensuring better global coherence. But this also makes the search and inference process more complicated. Another option is again to use ILP. An interesting property of ILP is that it performs global inference based on the output of local models rather than formulating a new inference procedure for solving the basic task. As we will see, ILP allows us to add global constraints (e.g., transitivity constraints) to ensure better global coherence between the various pairwise coreference decisions.

5.2 Integer Linear Programming

In this section, we give a very brief overview of the framework of ILP (see Cormen et al. (2001) for a detailed presentation). Developed during the second world war, linear programming (LP) is a well-known optimization technique that is now used by many industries (e.g., airline companies) in their daily planning. Its invention is generally attributed to three mathematicians: George B. Dantzig, John von Neumann, and Leonid Kantorovich.

In its *standard form*, a LP problem is an optimization problem consisting of two main parts:

- an objective function (to maximize) that can be specified as a *linear* function of certain variables: $c_1x_1 + c_2x_2 + \ldots c_nx_n$ (where c_i are assignment costs)
- problem constraints can be formulated as *equalities* or *inequalities* on those variables:

$$a_1x_1 + a_2x_2 + \ldots a_nx_n \leq b$$

An ILP problem is an LP problem in which all the unknown variables are required to be integers.

Geometrically, the linear constraints define a *convex polyhedron*, called the *feasible region*. Since the objective function is also linear, hence a convex function, all local optima are automatically global optima. An example of a simple ILP problem and its feasible region is provided in Figure 5.1.

maximize: $x_1 + x_2$

subject to: $x_1, x_2 \geq 0$
$x_1 + 2x_2 \leq 10$
$4x_2 - x_1 \leq 8$
$5x_1 - 2x_2 \leq -2$

solutions: $x_1 = 6; x_2 = 2$

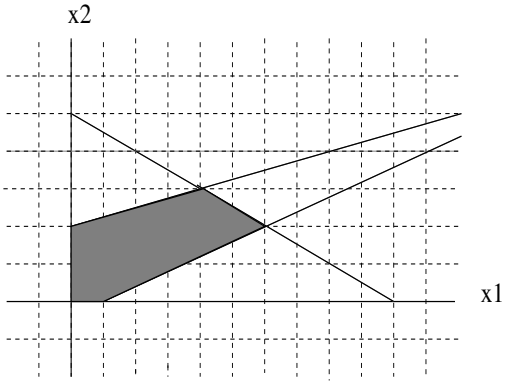

Figure 5.1: A linear program with two variables

Various methods have been developed for solving LP problems, but the most well-known method is the Simplex algorithm, originally developed by Dantzig. Very roughly, this algorithm solves LP problems by constructing an admissible solution at a vertex of the feasible region, and then walking along edges of that region to vertices with successively higher values of the objective function until the optimum is reached. Although quite efficient in practice, this algorithm has a poor worst case complexity, since it is polynomial. ILP problems are worse since they are NP-hard when they utilize bounded variables.

ILP formulations have a number of advantages for NLP problems. They are very expressive in allowing to represent many types of constraints in declarative fashion. They are also optimal: one is always guaranteed to find the optimal solution. And despite their

complexity, they are usually very fast: existing packages (e.g., CPLEX, GLPK, LPSOLVE)[3] are able to quickly solve very large problems. Previous uses of ILP in NLP include Roth and Yih (2004), Barzilay and Lapata (2006), Clarke and Lapata (2006), Riedel and Clarke (2006).

5.3 Base models

5.3.1 The coreference classifier

The first base model we use is a standard pairwise coreference classifier as described in Chapter 2. That is, this classifier determines for any pair of mentions $\langle i, j \rangle$ whether i and j coreferential or not. That is, this model estimates the probability $P_{scc}(\text{COREF}|\langle i, j \rangle)$. The construction and the application of this model follow the description from the previous chapter, and we will here refer to it as **COREF-PAIRWISE**. Specifically, this classifier was modeled using log-linear models, and the creation of the training instances follows the method described by Ng and Cardie (2002a). The feature set is also the same as the one used in Chapter 4. During testing, this model uses a "Best-First" link selection mechanism: that is, for each anaphor j, the predicted antecedent is the mention associated with the positive test instance that receives the highest score.

5.3.2 The discourse status classifier

A large number of errors made by coreference systems such as the one presented in Section 5.3.1 actually originate in errors in determining the discourse status of mentions. Thus, numerous errors come from when: (i) the system mistakenly identifies an antecedent for non-anaphoric mentions, and (ii) the system does not try to resolve an actual anaphoric mention. One way to counter such problem is to augment the coreference resolution system with a separate classifier which is used to determine the discourse status of mentions.

[3] CPLEX is a commercial software, while the later two are open-source. These can be found at: `http://www.gnu.org/software/glpk/` and `http://lpsolve.sf.net/`.

The discourse status model we used here follows the description given in Chapter 4, where the model was used in combination with specialized rankers. That is, discourse status determination is treated as binary classification problem with two outcomes: OLD and NEW. Through training, the classifier estimates the conditional probability $P_{ds}(c|i)$, where $c \in \{\text{OLD}, \text{NEW}\}$ and i is a mention. This probability model is based on log-linear models. The training procedure and the feature set for this model have been detailed in Chapter 4. As noted, the discourse status model achieves an overall accuracy score of 80.8% on the entire ACE dataset. The results for the model trained/tested on the different datasets are as follows: 80.1 for BNEWS, 82.2 NPAPER, and 81.1 for NWIRE.

5.3.3 The named entity classifier

In contrast to the previous binary tasks, named entity classification involves 5 class labels (namely, the ACE labels). The set of named entity type $\mathcal{T}$ are: FACility, GPE (geo-political entity), LOCation, ORGanization, PERson. The classifier estimates the conditional probabilities $P_{ne}(t|i)$ for each $t \in \mathcal{T}$ and predicts the named entity type $\widehat{t}$ for i such that $\widehat{t} = argmax_{t \in \mathcal{T}} P_{ne}(t|i)$.

$$P_{ne}(t|i) = \frac{\exp \sum_{j=1}^{m} \lambda_j f_j(i,t)}{\sum_{t'} \exp \sum_{j=1}^{m} w_j f_j(i,t')} \tag{5.1}$$

The same development cycle as described for the other models was used for this model. The features for named entity classification include: (i) the string of the mention, (ii) features defined over the string (e.g., whether it is capitalized, whether it contains punctuation, the head word), (iii) features describing the word and POS context around the mention, (iv) the Wordnet senses (including all the senses in the hypernym closure) associated with the head word of the mention. The feature set is described in more detail in Table 5.1.

String-based	
full_str	the entire string for i
hd_wd	the head word in i
first_wd	the first word in i
all_caps	all the letters in i are capitalized $\{1,0\}$
all_caps_periods	the string for i is a mixed of caps and periods $\{1,0\}$
starts_with_cap	the first letter in i is capitalized $\{1,0\}$
comma	the string for i contains a comma $\{1,0\}$
Context	
left_pos	POS of the token preceding i
right_pos	POS of the token following i
surr_pos	pair of POS for the tokens surrounding i
left_wd	word token preceding i
right_wd	word token following i
surr_wd	word tokens surrounding i
Wordnet	
wn_sense	Wordnet senses for i

Table 5.1: Feature selection for the named entity classifier

The named entity classifier achieves 79.5% on the ENTIRE ACE corpus (BNEWS: 79.8, NPAPER: 73.0, NWIRE: 72.7).

5.4 Base model results

This section describes the performance of the pairwise coreference classifier, both when used alone (**COREF-PAIRWISE**) and when used in a cascade with: (i) the discourse status classifier acting as a filter on which mentions should be resolved (**DS-CASCADE**), (ii) the named entity classifier acting as a filter on which mentions should be considered as antecedent candidates during resolution (**NE-CASCADE**), (iii) the two classifiers acting as combined filters (**DS-NE-CASCADE**).[4]

We also provide results for the corresponding oracle systems: (i) **ORACLE-DS** has perfect knowledge about discourse status (i.e., only true anaphors are resolved), (ii)

[4]The **DS-CASCADE** system corresponds to the **SCC+DS** in the previous chapter.

System	MUC			B^3			CEAF		
	R	P	F	R	P	F	R	P	F
COREF-PAIRWISE	60.8	72.6	66.2	62.4	77.7	69.2	62.3	62.3	62.3
DS-CASCADE	64.9	72.3	68.4	65.6	74.1	69.6	63.4	63.4	63.4
NE-CASCADE	56.3	75.2	64.4	59.6	82.4	69.2	61.6	61.6	61.6
DS-NE-CASCADE	61.3	68.8	64.8	62.5	73.8	67.7	61.9	61.9	61.9
ORACLE-DS	75.6	75.6	75.6	71.4	70.7	71.1	71.5	71.5	71.5
ORACLE-NE	62.5	81.3	70.7	62.9	85.5	72.4	65.2	65.2	65.2
ORACLE-DS-NE	83.2	83.2	83.2	79.0	78.2	78.6	78.7	78.7	78.7

Table 5.2: Recall (R), precision (P), and f-score (F) using MUC, B^3, and CEAF on the entire ACE corpus for the basic coreference system, the cascade systems, and the corresponding oracle systems.

ORACLE-NE has perfect knowledge about named entities (i.e., only mentions of the same entity than the current "anaphor" are considered as candidates), (iii) **ORACLE-DS-NE** has perfect knowledge about both discourse status and named entities.

Table 5.2 summarizes the results in terms of recall (R), precision (P), and f-score (F) on the three coreference metrics: MUC, B^3, and CEAF, respectively. Some overarching patterns emerge from these results. The first thing to note is the use of the cascade models in general fails to produce significant overall f-score improvements over the pairwise model **COREF-PAIRWISE**. These systems are far behind in performance from their corresponding oracles. This tendency is even stronger when the two filter models are applied, since **DS-NE-CASCADE** does significantly worse than **COREF-PAIRWISE**. In fact, this system has the lowest f-scores on the B^3 evaluation metric, suggesting that the errors of the two filters accumulate in this case. Note, on the other hand, that the combined oracle **ORACLE-DS-NE** achieves the best overall f-score results. It does so by capitalizing on the improvements given by the separate oracles. This oracle model shows large recall and precision improvements. The overall f-scores for these systems are as follows: 83.2% with MUC, 78.6% with B^3, and 78.7% with CEAF.

Secondly, note that the use of the two auxiliary models have complementary effects on the MUC and B^3 metrics, in both the cascade and the oracle systems. Thus, the use of the

discourse status classifier leads to recall improvements (suggesting that some true anaphors get "rescued" by this model), while the the use of the named entity model leads to precision improvements (suggesting that this model manages to filter out incorrect candidates that would have been chosen by the coreference model). In the case of the oracle systems, these gains translate in overall f-score improvements. But, as noted, this is generally not the case with the cascade systems. Only **DS-CASCADE** shows significant gains with MUC and CEAF (and not with B^3). **NE-CASCADE** underperforms in all three metrics. This later system indeed shows important drops in recall, suggesting that this model filter is overzealous in filtering true antecedents.

5.5 Integer programming formulations

This section provides several ILP formulations for coreference resolution. The first formulation **COREF-ILP** is based on the coreference classifier alone, and will serve as a baseline for evaluating the other, joint formulations. This first model allows a single anaphor to take multiple antecedents (in contrast with usual single-link clustering algorithms). Technically, this formulation does not require ILP, and it is equivalent to using the "Aggressive-Merge" clustering of McCarthy and Lehnert (1995). The other formulations provide joint inference over groups of base models. **JOINT-DS-ILP** combines the coreference classifier with the discourse status classifier, **JOINT-NE-ILP** combines it with the named entity classifier, and **JOINT-DS-NE-ILP** combines all three. For each joint formulation, *consistency* constraints ensure that the ultimate assignments for each task are mutually consistent. Finally, we describe the use of additional global constraints on coreference decisions; these are applicable to all of the formulations.

For solving the ILP problem, we use CPLEX, a commercial LP solver which implements the Simplex and the Branch-and-Bound methods. In practice, each document is processed to define a distinct ILP problem that is then submitted to the solver.

5.5.1 COREF-ILP: coreference-only formulation

COREF-ILP uses an objective function based on *only* the coreference classifier and the probabilities it produces. From the output probabilities $p_C = P_C(\text{COREF}|i,j)$, we define the assignment cost of committing to a coreference link as $c^C_{\langle i,j\rangle} = -log(p_C)$. The complement assignment cost of choosing not to establish a link is: $\overline{c}^C_{\langle i,j\rangle} = -log(1-p_C)$. $\mathcal{M}$ denotes the set of mentions, and $\mathcal{P}$ the set of possible coreference links over these mentions (i.e., $\mathcal{P} = \{\langle i,j\rangle | \langle i,j\rangle \in \mathcal{M} \times \mathcal{M} \text{ and } i < j\}$). Finally, we use indicator variables $x_{\langle i,j\rangle}$ that are set to 1 if i and j corefer, and 0 otherwise. The objective function takes the form:

$$min \sum_{\langle i,j\rangle \in \mathcal{P}} c^C_{\langle i,j\rangle} \cdot x_{\langle i,j\rangle} + \overline{c}^C_{\langle i,j\rangle} \cdot (1 - x_{\langle i,j\rangle}) \tag{5.2}$$

subject to:

$$x_{\langle i,j\rangle} \in \{0,1\} \qquad\qquad \forall \langle i,j\rangle \in \mathcal{P}$$

This formulation is similar to that of (Barzilay and Lapata, 2006); these authors use ILP for the problem of aggregating propositions for NL generation. But note that we minimize rather than maximize due to the fact we transform the model probabilities with $-log$ (like (Roth and Yih, 2004)).

This objective function on its own simply guarantees that ILP will find a global assignment that maximally agrees with the decisions of the coreference classifier. This actually amounts to taking all links for which the classifier returns a probability above .5; as noted, this is strictly equivalent to the "Aggressive-Merge" clustering of McCarthy and Lehnert (1995).[5]

[5]It is worth noting that the "Best-First" clustering can be simulated within ILP in the form of a constraint requiring that each mention j is linked to at most one mention i. Since we are maximizing, this indeed amounts to take the antecedent with the highest score.

5.5.2 JOINT-DS-ILP: joint discourse status-coreference formulation

The **JOINT-DS-ILP** system brings the two decisions of discourse status and coreference together by including both in a single objective function and including constraints that ensure the *consistency* of a solution for both tasks. Let c_j^A and $\overline{c}_j^A$ be defined analogously to the coreference classifier costs for $p_A = P_{ds}(\text{OLD}|j)$, the probability the discourse status classifier assigns to a mention j being anaphoric (i.e., discourse-old). Also, we have indicator variables y_j that are set to 1 if mention j is anaphoric and 0 otherwise. The objective function takes the following form:

$$min \sum_{\langle i,j\rangle \in \mathcal{P}} c^C_{\langle i,j\rangle} \cdot x_{\langle i,j\rangle} + \overline{c}^C_{\langle i,j\rangle} \cdot (1 - x_{\langle i,j\rangle}) + \sum_{j \in \mathcal{M}} c_j^A \cdot y_j + \overline{c}_j^A \cdot (1 - y_j)$$

subject to:

$$x_{\langle i,j\rangle} \in \{0,1\} \qquad \forall \langle i,j\rangle \in \mathcal{P}$$
$$y_j \in \{0,1\} \qquad \forall j \in \mathcal{M}$$

This function does not constrain the assignment of the $x_{\langle i,j\rangle}$ and y_j variables to be consistent with one another. To enforce consistency, we add further constraints. In what follows, $\mathcal{M}_j$ is the set of all mentions preceding mention j in the document.

Resolve all anaphors: if a mention is anaphoric (y_j=1), it *must* have at least one antecedent.

$$y_j \leq \sum_{i \in \mathcal{M}_j} x_{\langle i,j\rangle} \qquad \forall j \in \mathcal{M}$$

Resolve only anaphors: if a pair of mentions $\langle i,j\rangle$ is coreferent ($x_{\langle i,j\rangle}$=1), then j

is anaphoric (y_j=1).

$$x_{\langle i,j \rangle} \leq y_j \qquad\qquad \forall \langle i,j \rangle \in \mathcal{P}$$

These constraints thus directly relate the two tasks. By formulating the problem this way, the decisions of the discourse status classifier are not taken on faith as they were with **DS-CASCADE**. Instead, we optimize over consideration of both possibilities in the objective function (relative to the probability output by the classifier) while ensuring that the final assignments respect the significance of what it is to be anaphoric or non-anaphoric. Note that the effect of these two constraints remains in a sense local since they leave the possibility of "implicit" anaphors. By that, we mean cases in which the final discourse status assignment for a mention j says it isn't anaphoric (i.e., $y_j j = 0$), but j is in fact anaphoric as a result of transitive closure (e.g., if $x_{\langle i,k \rangle} = x_{\langle j,k \rangle} = 1$). Such a situation is now possible, since more than one antecedent is allowed per anaphor. This type of case motivates the use of additional constraints relating pairs of assignments; these are discussed in Section 5.5.5.

5.5.3 JOINT-NE-ILP: joint entity-coreference formulation

In this second joint formulation, we combine coreference decisions with named entity classification. New indicator variables for the assignments of this model are introduced, namely $z_{\langle i,j \rangle}$, where $\langle i,t \rangle \in \mathcal{M} \times \mathcal{T}$. Since entity classification is not a binary decision, each assignment variable encodes a mention i and a named entity type t. Each of these variables have an associated cost $c^E_{\langle i,t \rangle}$, which is the probability that mention i has type t: $c^E_{\langle i,t \rangle} = P_E(t|i)$. The objective function for this formulation is:

$$min \sum_{\langle i,j \rangle \in \mathcal{P}} c^C_{\langle i,j \rangle} \cdot x_{\langle i,j \rangle} + \overline{c}^C_{\langle i,j \rangle} \cdot (1 - x_{\langle i,j \rangle}) + \sum_{\langle i,t \rangle \in \mathcal{M} \times \mathcal{T}} c^E_{\langle i,t \rangle} \cdot z_{\langle i,t \rangle}$$

subject to:

$$z_{\langle i,t\rangle} \in \{0,1\} \qquad \forall \langle i,t\rangle \in \mathcal{M} \times \mathcal{T}$$

$$\sum_{i \in \mathcal{M}} z_{\langle i,t\rangle} = 1 \qquad \forall i \in \mathcal{M}$$

The last constraint ensures that each mention is only assigned a unique named entity type. Consistency between the two models is ensured with the constraint:

Coreferential mentions have the same entity type: if i and j are coreferential ($x_{\langle i,j\rangle}$=1), then they must be have the same type ($z_{\langle i,t\rangle} - z_{\langle j,t\rangle} = 0$):

$$1 - x_{\langle i,j\rangle} \geq z_{\langle i,t\rangle} - z_{\langle j,t\rangle} \qquad \forall \langle i,j\rangle \in \mathcal{P}, \forall t \in \mathcal{T}$$

$$1 - x_{\langle i,j\rangle} \geq z_{\langle j,t\rangle} - z_{\langle i,t\rangle} \qquad \forall \langle i,j\rangle \in \mathcal{P}, \forall t \in \mathcal{T}$$

These constraints above make sure that the coreference decisions are informed by the named entity classifier and vice versa. Furthermore, because these constraints ensure like assignments to coreferent pairs of mentions, they have a chaining effect that makes the overall system global. Coreference assignments that have low cost (i.e., high confidence) can influence named entity assignments (e.g., from a COMPANY to a PERSON). This in turn can alter other coreference assignments involving further mentions radiating out from one core, highly likely assignment.

5.5.4 JOINT-DS-NE-ILP: joint discourse status-entity-coreference formulation

For the third joint model, we combine all three base models with an objective function that is the composite of those of **JOINT-DS-ILP** and **JOINT-NE-ILP** and incorporate all the constraints that go with them. By creating a triple joint model, we get constraints between discourse status and named entity classification for free, as a result of the interaction of the consistency constraints between discourse status and coreference and of those between

named entity and coreference. For example, if a mention of type t is anaphoric, then there must be at least one mention of type t preceding it.

5.5.5 Transitivity constraints

The different ILP formulations given above can be further extended by a number of global constraints, i.e. constraints that use a larger context. Inspired by (Barzilay and Lapata, 2006), the constraints we propose exploit the fact that coreference is an equivalence relation. Thus, one can constrain triples of mentions i, j, k, where $i < j < k$ using the following three constraints on coreference assignments. These constraints in effect account for the dependencies between the different coreference decisions. In what follows, $M_{i,j,k}$ is the set of triples $\langle i, j, k\rangle$ such that $\langle i, j, k\rangle \in \mathcal{M} \times \mathcal{M} \times \mathcal{M}$ and $i < j < k$.

Transitivity: if $x_{\langle i,j\rangle}$ and $x_{\langle j,k\rangle}$ are coreferential pairs (i.e., $x_{\langle i,j\rangle} = x_{\langle j,k\rangle} = 1$), then so is $x_{\langle i,k\rangle}$:

$$x_{\langle i,k\rangle} \geq x_{\langle i,j\rangle} + x_{\langle j,k\rangle} - 1 \qquad \forall\langle i, j, k\rangle \in M_{i,j,k}$$

Euclideanity: if $x_{\langle i,k\rangle}$ and $x_{\langle j,k\rangle}$ are coreferential pairs (i.e., $x_{\langle i,k\rangle} = x_{\langle j,k\rangle} = 1$), then so is $x_{\langle i,j\rangle}$:

$$x_{\langle i,j\rangle} \geq x_{\langle i,k\rangle} + x_{\langle j,k\rangle} - 1 \qquad \forall\langle i, j, k\rangle \in M_{i,j,k}$$

Anti-Euclideanity: if $x_{\langle i,j\rangle}$ and $x_{\langle i,k\rangle}$ are coreferential pairs (i.e., $x_{\langle i,j\rangle} = x_{\langle i,k\rangle} = 1$), then so is $x_{\langle j,k\rangle}$.

$$x_{\langle j,k\rangle} \geq x_{\langle i,j\rangle} + x_{\langle i,k\rangle} - 1 \qquad \forall\langle i, j, k\rangle \in M_{i,j,k}$$

Enforcing the latter constraint alone guarantees that the final assignment will not produce any implicit anaphor (and no chain will have a mention predicted to be anaphoric as their head). The interaction of this constraint with **resolve only anaphors** guarantees that the three assignments $x_{\langle j,k \rangle} = 1$, $x_{\langle i,k \rangle} = 1$, and $y_j = 0$ cannot all together be part of the final global assignment.

Note that one could have one unique transitivity constraint if we had symmetry in our model; concretely, capturing symmetry means: (i) adding a new indicator variable $x_{\langle j,i \rangle}$ for each variable $x_{\langle i,j \rangle}$, and (ii) making sure $x_{\langle j,i \rangle}$ agrees with $x_{\langle i,j \rangle}$.

Enforcing each of these constraints above means adding $\frac{1}{6} \times n \times (n-1) \times (n-2)$ constraints, for a document containing n mentions. This means close to $500,000$ of these constraints for a document containing just 100 mentions. The inclusion of such a large set of constraints turned out to be difficult, causing memory issues with large documents (some of the ACE documents have more than 250 mentions). Consequently, we investigated during development various simpler scenarios, such as enforcing these constraints for documents that had a relatively small number of mentions (e.g., 100) or just using one of these types of constraint (in particular **Anti-Euclideanity** given the way it interacts with the discourse status assignments). In the following, **JOINT-DS-NE-AE-ILP** will refer to the **JOINT-DS-NE-ILP** formulation augmented with the **Anti-Euclideanity** constraints.

5.5.6 Other global constraints

Transitivity captures dependencies between coreference decisions by imposing coherence on triples of mentions. Below, we suggest two other types of constraint that can be imposed on the whole partitioning. Note however that these constraints have not yet been included in any of our ILP formulations. The first constraint controls the overall number of anaphors in the document; this is achieved by providing a lower bound α on the number of discourse status assignments:

System	MUC			B^3			CEAF		
	R	P	F	R	P	F	R	P	F
COREF-PAIRWISE	60.8	72.6	66.2	62.4	77.7	69.2	62.3	62.3	62.3
COREF-ILP	70.3	72.7	71.5	73.2	63.7	68.1	58.7	58.7	58.7
JOINT-DS-ILP	73.2	73.4	73.3	75.3	62.0	68.0	58.9	58.9	58.9
JOINT-NE-ILP	66.2	75.0	70.4	69.6	71.2	70.4	61.2	61.2	61.2
JOINT-DS-NE-ILP	69.6	75.4	72.4	72.2	69.7	70.9	62.3	62.3	62.3
JOINT-DS-NE-AE-ILP	63.7	77.8	70.1	65.6	81.4	72.7	66.2	66.2	66.2

Table 5.3: Recall (R), precision (P), and f-score (F) using the MUC, B^3, and CEAF evaluation metric on the the entire ACE dataset for the ILP coreference systems.

$$\sum_{j \in \mathcal{M}} y_j \leq \alpha \tag{5.3}$$

If used with the transitivity constraints, this constraint can be seen as constraining the total number of entities in the document. This is because the number of entities is the same as the number of non-anaphors: each non-anaphor corresponds to one distinct entity. The second constraint controls the number of coreference links for a given anaphor, by putting a lower bound λ:

$$\sum_{\langle i,j \rangle \in \mathcal{P}} x_{\langle i,j \rangle} \leq \lambda \tag{5.4}$$

The two parameters α and λ can be estimated on development data for each of the dataset. Note that the estimation of λ depends on whether or not the transitivity constraints are also enforced and the window that is used. This is because transitivity will affect the overall number of links.

5.6 ILP results

Table 5.3 summarizes the scores for the different ILP systems for MUC, B^3, and CEAF, respectively. The first thing to notice is that two ILP formulations that combine the three

local models, namely **JOINT-DS-NE-ILP** and **JOINT-DS-NE-AE-ILP**, deliver significant improvements over both **COREF-PAIRWISE** and **COREF-ILP** on the three evaluation metrics. In fact, these two formulations provide the best f-scores on both the B^3 and CEAF: the gains on these metrics go as high as 3.5% over **COREF-PAIRWISE** and 4.6% over **COREF-ILP**, while the gains on CEAF are of 3.9% over **COREF-PAIRWISE** and 7.3% over **COREF-ILP**. On MUC, **JOINT-DS-NE-ILP** is the second best performing system, only after **JOINT-DS-ILP**.

These results are in sharp contrast with those obtained by the cascade model **DS-NE-CASCADE**: recall that this system, while also using the two auxiliary models, was worse than **COREF-PAIRWISE**. They clearly show the superiority of the joint formulation over the cascade approach for integrating and combining the extra information provided by the discourse status and named entity models. In addition to improving coreference resolution performance, the joint formulations also yield improvement on the named entity classification: specifically, accuracy for that task went from 79.5% to over 80% for each of the ILP formulations using this model. [6]

Although the ILP formulations perform better overall, they show different patterns of results depending on the different evaluation metrics. In particular, there is a clear split between MUC and the other two metrics, which is based on two important differences. First, the simple ILP formulation, **COREF-ILP**, performs comparatively much better in MUC than in B^3 and CEAF. This system already significantly outperforms **COREF-PAIRWISE** on MUC (with gains of 5.3%), but it does worse than **COREF-PAIRWISE** on the two other metrics. Second, the ILP formulations that incorporate the named entity model and the anti-euclidean constraints fail to provide improvements over the simpler formulations **COREF-ILP** and **JOINT-DS-ILP** in MUC, while they are the best systems in B^3 and CEAF. These differences can actually be traced back to the way the different metrics work. In particular, recall that MUC favors systems that produce a large number of coreference links (by the same token, it

[6] Accuracy for discourse status goes down, from 80.9% to 80.0% on the entire ACE corpus.

is more lenient with systems that have poor precision). This bias first explains why **COREF-ILP** does so much better than **COREF-PAIRWISE**: the only difference between these two systems is indeed that **COREF-ILP** produces more links by allowing more than one antecedent per anaphor. The important recall improvements given by **COREF-ILP** directly translate in f-score gains on the MUC metric, but not on B^3 and CEAF which both strongly penalize this system in terms of precision. Similarly, only these two metrics show the benefits of including the named entity model and the anti-euclidean constraints. These provide important precision improvements which, combined with the recall gains provided by the inclusion of the discourse status model, are able to yield overall f-score improvements.

Further experiments reveal that bringing the other transitivity constraints into the ILP formulation results in additional *precision* gains, although not in overall f-score gains. The effect of these constraints is indeed of withdrawing incoherent links, rather than producing new links. At the global level, this results in the creation of smaller, more coherent clusters of mentions. Switching on these constraints may therefore be useful for certain applications where precision is more important than recall. Finally, we expect that the addition of the other global constraints, which control the shape of the whole partitioning, will be able to better balance recall and precision.

5.7 Summary and discussion

In this chapter, we have provided a new approach to the task of coreference resolution by recasting it as a linear optimization problem. In particular, we have used the framework of ILP to cleanly integrate the predictions of three different local models (namely, a standard pairwise coreference classifier, a discourse status classifier, and a named entity classifier) and to perform global inference over these models. Our ILP formulations cleanly capture the dependencies between these different models through the use of simple declarative constraints which mutually constrain the final outcomes of the models. Crucially, this means that optimization is achieved without careful weighting of the models. In addition, we

have also shown how to incorporate various other global constraints (such as transitivity constraints) to better capture the dependencies between coreference decisions.

In terms of performance, we have demonstrated that the various ILP formulations provide overall f-score improvements over both the standard pairwise model and the cascade models. Improvements were found across the three different evaluation metrics: MUC B^3, and CEAF. The fact that B^3 and CEAF scores were also improved is significant: the ILP formulations tend to construct larger coreference chains —these are rewarded by MUC without precision penalties, but B^3 and CEAF are not as lenient. Improvements in these two metrics thus give stronger evidence that the joint ILP formulations really do deliver better coreference assignments.

There are several natural extensions to the approach proposed here. Given the flexibility of the ILP framework in integrating different models, a first way to extend this approach is to include other coreference models like the specialized models described in Chapter 4. While the integration of additional classifiers is straightforward, the integration of the rankers is more complicated. The difficulty has to do with the fact that rankers provide a different type of probability distribution than classifiers: that is, they provide a probability distribution over the set of antecedent candidates for a given anaphor. This raises the question of how to best convert these probabilities in terms of assignment costs.[7]

Another, potentially fruitful way to extend this approach is to incorporate other types of model. In particular, we are very interested in combining coreference models and discourse parsing models. Discourse theories, e.g., Asher and Lascarides (2003), have for a long time emphasized the interdependence of the two problems: coreference plays an important role in establishing discourse coherence, and coherence also plays an important in constraining reference resolutions. ILP would provide a very suitable framework for modeling these two tasks as a joint problem.

[7]One straightforward method would be to use the probability given to each candidate by the ranker as the cost for making a link to that particular candidate, and to use the uniform probability (i.e., 1 over the total number of candidates) as the cost for not making the link. But a potential problem with this approach is that the size of the candidate sets varies from one anaphor to the other: this is likely to make these costs unreliable.

Chapter 6

Conclusions

The main goal of this dissertation has been to investigate and develop more effective learning models for robust anaphora and coreference resolution. As our starting point, we identified in Chapter 2 four potential limitations inherent to the existing approaches to these tasks. The first limitation regards the **type of model** that has been being used. The vast majority of previous learning-based systems recasts reference resolution as binary classification, whereby each pair of nominal mentions is classified as either coreferential or not. The fundamental problem with this view is that it embeds an unwarranted independence assumption, namely that establishing reference between an anaphor and an antecedent candidate is independent from the other candidates. A second limitation of most existing approaches, specifically when dealing with coreference resolution, is that they construct a **single model** for resolving different referring expressions. This is problematic since different linguistic expressions (e.g., pronouns and proper names, to take two extremes) show different linking strategies —an observation made for a long time by both semanticists and psycholinguists. Another weakness of classification-based approaches to coreference regards the way the different coreference resolutions are coordinated. Typically, previous approaches have used extremely greedy clustering algorithms for merging the pairwise decisions, in effect treating **each linking decision as a purely local decision**. This is again an unwarranted assumption,

since the decision of merging a mention into a chain should be conditioned on how well it matches the entity as a whole. Finally, existing approaches have for the most part failed to exploit the rich knowledge sources necessary to properly model coreference. Crucially, attempts at **predicting and incorporating linguistically relevant information** (as features or pre- or post-processing modules) have been for the most part unsuccessful.

The main contribution of this thesis has been in developing a set of techniques —ranking models and integer linear programming— that are able to overcome the above limitations while remaining easy to design and computationally tractable. Common to these techniques is that they make fewer independence assumptions, and consider a more global context for making their decisions.

More specifically, we have first investigated the use of a **ranker** as an alternative to the traditional classification approach for the restricted task of pronoun resolution. As discussed, the ranker provides a better model of the problem of antecedent selection by directly bringing the comparison between the candidates inside the training criterion (rather than deriving it from the classifier's probabilities, which give only an imperfect estimation of antecedent-hood). The ranking approach for pronoun resolution yields large improvements (up to 8%) over the traditional pairwise classification model. The ranker also compares very favorably to the to-date best pronoun resolution system, namely the so-called twin-candidate approach of Yang et al. (2005), with an improvement of 1-2%. An important advantage of the ranker over this later model is that the ranker has much faster training and online testing times: specifically, the complexity of the twin-candidate model is cubic in the number of mentions in a document, while that of the ranker is only square.

Second, we have also shown that ranking works well for full coreference resolution, as long as: (i) one is able to reliably filter out discourse-new mentions (i.e., the non-anaphors), and (ii) the resolution task is split into different models for different linguistic expressions. Thus, the use of rankers was extended from pronoun resolution to full coreference resolution through the creation of **specialized rankers** that deal with clearly defined

subsets of the coreference problem: third-person pronouns, speech pronouns, proper names, definite descriptions, and all others. This "distributed" strategy led to 3-4% improvements in coreference f-score across three different evaluation metrics (MUC, B^3, and CEAF), compared to using a standard classification approach. Furthermore, when compared to using similarly specialized classifiers, the use of specialized rankers still led to significant improvements of 1-2%.

A considerable improvement over the classification-based approaches, the proposed ranking approach still suffers from two important shortcomings. By relying on a simple pipeline architecture, this approach first fails to model the dependencies between discourse status determination and coreference linking. Second, like the classification-based approaches, this approach makes each coreference decision independently of one another. For these reasons, we explore in Chapter 5 a drastically different view of the coreference problem, and recast it as an optimization problem. In particular, we use the framework of Integer Linear Programming (ILP) to combine different, locally trained models into a **joint, global inference problem**. The three models are: a standard pairwise coreference classifier, a discourse status classifier, and also a named entity classifier. The final predictions of the three models are mutually constrained through the use of simple, declarative constraints which capture the dependencies between the models. The ILP framework also allows us to integrate various other global constraints (such as transitivity constraints), whose role is to better capture the dependencies between coreference decisions. Tested on the ACE datasets, our ILP formulations deliver significant f-score improvements over both a standard pairwise model, and various models that employ the discourse status and a named entity classifiers in a cascade. Improvements of 3-6% were found across the three evaluation metrics. Schematically, the joint formulations resulted in recall improvements, while the addition of the transitivity constraints improved precision. The fact that B^3 and CEAF scores were also improved is of particular importance: the ILP formulations tend to construct larger coreference chains —these are rewarded by MUC without precision penalties,

but B^3 and CEAF are not as lenient. Improvements in these two metrics thus give strong experimental evidence that the joint ILP formulations really do deliver better coreference assignments.

There are a number of directions in which to extend the work presented in this direction. The most obvious is to try to bring in the different ranker models into the ILP formulation: given that they provide better local models than classifiers, one expects them to yield even better results when integrated in a global formulation. The second is in the integration of more numerous, and richer information sources. As witnessed in the learning curves of Chapter 3, surface-based features can only do so well, and cause learning to plateau after a relatively small number of documents. Among the most promising sources of information to add are deeper syntactic knowledge, lexical semantics, and discourse structure. Interestingly, the present work (in particular the ILP formulations) provides a very general and simple infrastructure in which additional knowledge sources (e.g., in the form of additional models) can be easily integrated. Finally, we would like to extend this research to other languages as well as to other related phenomena. Most of current work on coreference deals with nominal anaphora, and there is only very little research on computational treatments for abstract entity anaphora or temporal anaphora. Since there are few (if any) resources available in these two cases, we would like to explore unsupervised and/or semi-supervised techniques such as active learning, as well as domain adaptation techniques.

Bibliography

Proceedings of the 6th Message Understanding Conference (MUC-6), San Mateo, CA, 1995. Morgan Kaufmann.

Proceedings of the 7th Message Understanding Conference (MUC-7), `http://acl.ldc.upenn.edu/muc7`, 1998.

Mira Ariel. Referring and accessibility. *Journal of Linguistics*, pages 65–87, 1988.

Nicholas Asher. *Reference to Abstract Objects in Discourse*. Kluwer, 1993.

Nicholas Asher and Alex Lascarides. *Logics of Conversation*. Cambridge University Press, Cambridge, UK, 2003.

Nicholas Asher, Pascal Denis, and Brian Reese. Names and pops and discourse structure. In *Workshop on Constraints in Discourse*, Maynooth, Ireland, 2006.

S. Azzam, K. Humphreys, and R. Gaizauskas. Using coreference chains for text summarization. In *Proceedings of the ACL Workshop on Coreference and its Applications*, College Park, MD, 1999.

Amit Bagga and Breck Baldwin. Algorithms for scoring coreference chains. In *Proceedings of LREC*, pages 563–566, 1998.

Breck Baldwin. Cogniac: high precision coreference with limited knowledge and linguistic

resources. In *Proceedings of the ACL-EACL 2007 Workshop on Operational Factors in Practical, Robust Anaphora Resolution*, Madrid, Spain, 1997.

Regina Barzilay and Mirella Lapata. Aggregation via set partitioning for natural language generation. In *HLT-NAACL-06*, pages 359–366, New York City, USA, 2006.

D. Bean and E. Riloff. Unsupervised learning of contextual role knowledge for coreference resolution. In *HLT-NAACL-04*, 2004.

David Beaver. The optimization of discourse anaphora. *Linguistics and Philosophy*, 27(1), 2004.

A. Berger, S. Della Pietra, and V. Della Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71, 1996.

S. Brennan, M. Friedman, and C. Pollard. A centering approach to pronouns. In *Proceedings of ACL 1987*, pages 155–162, Stanford, CA, 1987.

Jaime G. Carbonell and Ralf D. Brown. Anaphora resolution: a multi-strategy approach. *Proceedings of COLING*, pages 96–101, 1988.

C. Cardie and K. Wagstaff. Noun phrase coreference as clustering. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 82–89, University of Maryland, MD, 1999. Association for Computational Linguistics.

David Carter. *Intepreting Anaphors in Natural Language Texts*. Ellis Horwood, Chichester, UK, 1987.

Eugene Charniak and Mark Johnson. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proceedings of ACL 2005*, Ann Arbor, Michigan, 2005.

Stanley F. Chen and Ronald Rosenfeld. A gaussian prior for smoothing maximum entropy models. Technical report, Carnegie Mellon University, 1999. Technical Report CMUCS -99-108.

Herbert H. Clark. Bridging. In R. C. Schank and B. L. Nash-Webber, editors, *Theoretical Issues in Natural Language Processing*. Association for Computing Machinery, New York, 1975.

James Clarke and Mirella Lapata. Constraint-based sentence compression: An integer programming approach. In *Proceedings of COLING/ACL*, pages 144–151, 2006.

Michael Collins and Nigel Duffy. New ranking algorithms for parsing and tagging: kernels over discrete structures and the voted perceptron. In *Proceedings of ACL 2002*, pages 263–270, Philadelphia, Pennsylvania, 2002.

Michael Collins and Terry Koo. Discriminative reranking for natural language parsing. *Computational Linguistics*, 31(1):25–70, 2005.

Dennis Connolly, John D. Burger, and David S. Day. A machine learning approach to anaphoric reference. In D. B. Jones and H. L. Somers, editors, *New Methods in Language Processing*, pages 133–153. UCL Press, 1997.

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT Press, second edition edition, 2001.

Dan Cristea, Nancy Ide, and Laurent Romary. Veins theory: A model of global discourse cohesion and coherence. In *Proceedings of ACL and the Seventeenth International Conference on Computational Linguistics*, pages 281–285, San Francisco, CA, 1998. Morgan Kaufmann Publishers.

Aron Culotta, Michael Wick, and Andrew McCallum. First-order probabilistic models for coreference resolution. In *Proceedings of HLT-NAACL 2007*, Rochester, NY, 2007.

I. Dagan and A. Itai. Automatic processing of large corpora for the resolution of anaphora references. In *Proceedings of COLING*, pages 330–332, 1990.

Pascal Denis and Jason Baldridge. A ranking approach to pronoun resolution. In *Proceedings of IJCAI 2007*, Hyderabad, India, 2007a.

Pascal Denis and Jason Baldridge. Joint determination of anaphoricity and coreference resolution using integer programming. In *Proceedings of HLT-NAACL 2007*, Rochester, NY, 2007b.

Pascal Denis and Jonas Kuhn. Applying an lfg-parser for coreference resolution: Experiments and analysis. In *Proceedings of the LFG conference*, Konstanz, Germany, 2006.

Thomas Dietterich. Ensemble methods in machine learning. In J. Kittler and F. Roli, editors, *First International Workshop on Multiple Classifier Systems*, Lecture Notes in Computer Science, pages 1–15, New York, 2000. Springer Verlag.

Barbara Grosz, Aravind Joshi, and Scott Weinstein. Centering: A framework for modelling the local coherence of discourse. *Computational Linguistics*, 2(21), 1995.

Barbara J. Grosz and Candace L. Sidner. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204, 1986.

Jeanette K. Gundel, Nancy Hedberg, and Ron Zacharski. Cognitive status and the form of referring expressions in discourse. *Language*, 69:274–307, 1993.

Aria Haghighi and Dan Klein. Unsupervised coreference resolution in a nonparametric bayesian model. In *Proceedings ACL*, pages 848–855, Prague, Czech Republic, 2007.

Lynette Hirschman and Nancy Chinchor. MUC-7 coreference task definition. In *Proceedings fo the 7th Message Understanding Conference (MUC-7)*, `http://acl.ldc.upenn.edu/muc7/co_task.html`, 1998.

Graeme Hirst. *Anaphora in Natural Language Understanding: A Survey*. Springer-Verlag, Berlin, Germany, 1981.

Jerry Hobbs. Pronoun resolution. Technical report, CUNY, 1976.

Jerry Hobbs. Resolving pronoun references. *Lingua*, 44:339–352, 1978.

Jerry Hobbs. Coherence and coreference. Technical report, SRI International, 1979.

Hans Kamp and Uwe Reyle. *From Discourse to Logic*. Kluwer, 1993.

A. Kehler, D. Appelt, L. Taylor, and A. Simma. The (non)utility of predicate-argument frequencies for pronoun interpretation. In *Proceedings of HLT-NAACL 2004*, 2004a.

A. Kehler, D. Appelt, L. Taylor, and A. Simma. Competitive self-trained pronoun interpretation. In *Proceedings of HLT-NAACL 2004*, 2004b.

Andrew Kehler. *Coherence, Reference, and the Theory of Grammar*. CSLI, 2002.

Andrew Kehler. Probabilistic coreference in information extraction. In *Proceedings EMNLP 1997*, pages 163–173, 1997.

C. Kennedy and B. Boguraev. Anaphora for everyone: Pronominal anaphora resolution without a parser. In *Proceedings of the 16th International Conference on Computational Linguistics*, pages 113–138, 1996.

Shalom Lappin and Herbert J. Leass. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–561, 1994.

Xiaoqiang Luo. On coreference resolution performance metrics. In *Proceedings of HLT-NAACL 2005*, pages 25–32, 2005.

Xiaoqiang Luo. Coreference or not: a twin model for coreference resolution. In *Proceedings of HLT-NAACL 2007*, pages 73–80, Rochester, NY, 2007.

Xiaoqiang Luo, Abe Ittycheriah, Hogyan Jing, Nanda Kambhatla, and Salim Roukos. A mention-synchronous coreference resolution algorithm based on the bell tree. In *Proceedings of ACL 2004*, pages 135–142, Barcelona, Spain, 2004.

Robert Malouf. A comparison of algorithms for maximum entropy parameter estimation. In *Proceedings of the Sixth Workshop on Natural Language Learning*, pages 49–55, Taipei, Taiwan, 2002.

A. McCallum and B. Wellner. Toward conditional models of identity uncertainty with application to proper noun coreference. In *Proceedings of IJCAI Workshop on Information Integration on the Web*, 2003.

Joseph F. McCarthy and Wendy G. Lehnert. Using decision trees for coreference resolution. In *IJCAI*, pages 1050–1055, 1995.

T. McEnery, I. Tanaka, and S. Botley. Corpus annotation and reference resolution. In *Proceedings of the ACL Workshop on Operational Factors In Practical, Robust, Anaphora Resolution for Unrestricted Texts*, pages 67–74, 1997.

Tom M. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

Ruslan Mitkov. *Anaphora Resolution*, pages 266–283. Oxford University Press, Oxford, 2002a.

Ruslan Mitkov. *Anaphora Resolution*. Longman, Harlow, UK, 2002b.

Ruslan Mitkov. Robust pronoun resolution with limited knowledge. In *Proceedings of COLING-ACL*, pages 869–875, 1998.

Thomas Morton. Coreference for NLP applications. In *Proceedings of ACL 2000*, Hong Kong, 2000.

Thomas Morton. Using coreference for question answering. In *Proceedings of ACL Workshop on Coreference and Its Applications*, 1999.

Vincent Ng. Machine learning for coreference resolution: Recent successes and future challenges. Technical report, Cornell University, 2002.

Vincent Ng. Learning noun phrase anaphoricity to improve coreference resolution: Issues in representation and optimization. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, 2004.

Vincent Ng. Supervised ranking for pronoun resolution: Some recent improvements. In *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05)*, 2005a.

Vincent Ng. Machine learning for coreference resolution: From local classification to global ranking. In *ACL-05*, pages 157–164, Ann Arbor, MI, 2005b.

Vincent Ng. Semantic class induction and coreference resolution. In *Proceedings of ACL*, 2007.

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

Vincent Ng and Claire Cardie. Identifying anaphoric and non-anaphoric noun phrases to improve coreference resolution. In *Proceedings of COLING 2002*, 2002b.

Miles Osborne and Jason Baldridge. Ensemble-based active learning for parse selection. In *Proceedings of HLT-NAACL 2004*, pages 89–96, Boston, MA, 2004.

Barabra Partee. Nominal and temporal anaphora. *Linguistics and Philosophy*, 7:243–286, 1984.

T. Pedersen, S. Patwardhan, and J. Michelizzi. WordNet::Similarity - Measuring the Relatedness of Concepts. In *Proceedings of the Nineteenth National Conference on Artifi-*

cial Intelligence (Intelligent Systems Demonstrations), pages 1024–1025, San Jose, CA, 2004.

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

Simone Paolo Ponzetto and Michael Strube. Exploiting semantic role labeling, WordNet and Wikipedia for coreference resolution. In *Proceedings of the HLT 2006*, pages 192–199, New York City, N.Y., 2006.

Andrei Popescu-Belis and Isabelle Robba. Three new methods for evaluating reference resolution. In *Proceedings of LREC 1998 Workshop on Linguistic Coreference*, Grenada, Spain, 1998.

Judita Preiss. A comparison of probabilistic and non-probabilistic anaphora resolution algorithms. In *Proceedings of the ACL Student Workshop*, pages 42–47, 2002.

Ellen F. Prince. Toward a taxonomy of given-new information. In P. Cole, editor, *Radical Pragmatics*, pages 223–255. Academic Press, New York, 1981.

A. Ratnaparkhi. *Maximum Entropy Models for Natural Language Ambiguity Resolution*. PhD thesis, University of Pennsylvania, 1998.

Deepak Ravichandran, Eduard Hovy, and Franz Josef Och. Statistical QA - classifier vs re-ranker: What's the difference? In *Proceedings of the ACL Workshop on Multilingual Summarization and Question Answering–Machine Learning and Beyond*. Association for Computational Linguistics, 2003.

Elaine Rich and Susann LuperFoy. An architecture for anaphora resolution. In *Proceedings of ACL*, pages 18–24, 1988.

Sebastian Riedel and James Clarke. Incremental integer linear programming for non-projective dependency parsing. In *Proceedings of EMNLP*, pages 137–137, 2006.

E. Riloff. An empirical study of automated dictionary construction for information extraction in three domains. *Artificial Intelligence*, 85:101–134, 1996.

Dan Roth and Wen-tau Yih. A linear programming formulation for global inference in natural language tasks. In *Proceedings of CoNLL*, 2004.

Candace L. Sidner. Focusing in the comprehension of definite anaphora. In M. Brady and R. Berwick, editors, *Computational Models of Discourse*, pages 267–330. MIT Press, 1983.

W. Soon, H. Ng, and D. Lim. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521–544, 2001.

Joel Tetreault. A corpus-based evaluation of centering and pronoun resolution. *Computational Linguistics*, 27(4):507–520, 2001.

Kristina Toutanova, Penka Markova, and Christopher Manning. The leaf projection path view of parse trees: Exploring string kernels for HPSG parse selection. In *EMNLP 2004*, pages 166–173, Barcelona, 2004.

Olga Uryupina. Linguistically motivated sample selection for coreference resolution. In *Proceedings of DAARC-2004*, Furnas, 2004.

Kees van Deemter and Rodger Kibble. On coreferring: Coreference in MUC and related annotation schemes. *Computational Linguistics*, 26(2):629–637, 2000.

Rob van der Sandt. Presupposition projection as anaphora resolution. *Journal of Semantics*, 9:333–377, 1992.

Erik Velldal and Stephan Oepen. Statistical ranking in tactical generation. In *Proceedings of EMNLP 2006*, Sydney, Australia, 2006.

R. Vieira and M. Poesio. An empirically-based system for processing definite descriptions. *Computational Linguistics*, 26(4):539–593, 2000.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. A model-theoretic coreference scoring scheme. In *Proceedings fo the 6th Message Understanding Conference (MUC-6)*, pages 45–52, San Mateo, CA, 1995. Morgan Kaufmann.

Marilyn Walker. Evaluationg discourse processing algorithms. In *Proceedings of ACL*, pages 251–261, 1989.

Bonnie Lynn Webber. *A Formal Approach to Discourse Anaphora*. Garland, New York, 1978.

X. Yang, G. Zhou, J. Su, and C.L. Tan. Coreference resolution using competitive learning approach. In *Proceedings of the ACL*, pages 176–183, 2003.

Xiaofeng Yang. *A Twin-Candidate Model for Learning Based Coreference Resolution*. PhD thesis, National University of Singapore, 2005.

Xiaofeng Yang, Jian Su, and Chew Lim Tan. Improving pronoun resolution using statistics-based semantic compatibility information. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 165–172, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.

Xiaofeng Yang, Jian Su, and Chew Lim Tan. Kernel-based pronoun resolution with structured syntactic knowledge. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 41–48, Sydney, Australia, July 2006. Association for Computational Linguistics.

Vita

Pascal Denis was born on June 7, 1974 in Saint-Mard, Belgium to Fernand Denis and Annie Toulmonde. He graduated from Athénée Royal de Virton in August 1992 and entered the Université Libre de Bruxelles the following Fall. He graduated with two Bachelor of Arts degrees: one in Journalism and Communication, received in September 1996, and the other in Romance Philology, received in September 1998. He subsequently earned a Master of Science in Cognitive Science from University of Edinburgh in September 1999, and a Master of Science in Artificial Intelligence from University of Toulouse in June 2000. Pascal entered the graduate program in Linguistics at the University of Texas in August 2000. He worked as a Teaching Assistant and a Graduate Research Assistant, and was awarded the Homer L. Bruce Fellowship in 2006.

Permanent Address: 81 Langton Street
Unit 8
San Francisco, CA 94103

This dissertation was typeset with LaTeX 2_ε[1] by the author.

[1] LaTeX 2_ε is an extension of LaTeX. LaTeX is a collection of macros for TeX. TeX is a trademark of the American Mathematical Society. The macros used in formatting this dissertation were written by Dinesh Das, Department of Computer Sciences, The University of Texas at Austin, and extended by Bert Kay, James A.

Bednar, and Ayman El-Khashab.