Machine Learning for Coreference Resolution: Recent Successes and Future Challenges

Vincent Ng
Department of Computer Science
Cornell University
Ithaca NY 14853
yung@cs.cornell.edu

November 26, 2002

Abstract

State-of-the-art coreference resolution systems are mostly knowledge-based systems that operate by relying on a set of hand-crafted coreference resolution heuristics. Recently, however, machine learning approaches have been shown to be a promising way to build coreference resolution systems that are more robust than their knowledge-based counterparts. Nevertheless, there are several key issues in existing machine learning approaches to the problem that are either not explored or being overlooked, potentially leading to a deterioration of system performance. This document examines each of these issues in detail and suggests potential solutions.

1 Introduction

Noun phrase coreference resolution refers to the problem of determining which noun phrases (NPs) refer to each real-world entity mentioned in a document. Specifically, given an arbitrary text as input, a coreference resolution system partitions the noun phrases in a text into coreference equivalence classes, with one cluster for each set of coreferent NPs. In the example shown in Figure 1, a coreference system should partition the noun phrases in the paragraph into five equivalence classes, generating three coreference chains — QE (*Queen Elizabeth* and *her*), KG (*husband*, *King George VI*, *the King*, and *his*) and LL (*Lionel Logue* and *a renowned speech therapist*) — and two NP singletons, *a viable monarch* and *speech impediment*. Hence, coreference is an equivalence relation defined on the set of noun phrases.[1] While human audiences have few problems of identifying co-referring NPs, coreference resolution can present a challenge to natural language processing (NLP) systems.

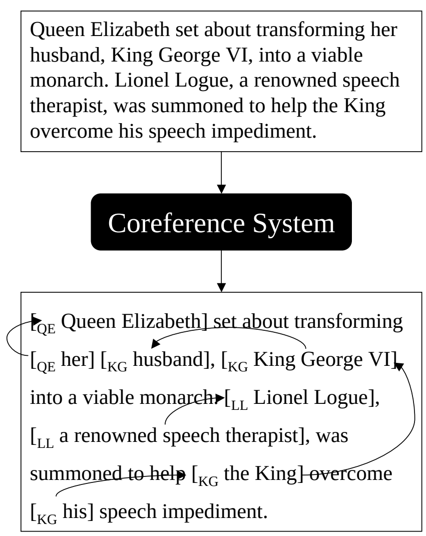

Figure 1: Coreference Resolution System

1.1 Anaphora and Coreference

Coreference resolution bears close resemblance to anaphora resolution, a classic NLP problem whose goal is to identify an antecedent for each noun phrase that depends on other noun phrases for its interpretation. To be precise, an NP refers independently (and hence does not depend on other NPs for its interpretation) if and only if it "selects" or uniquely identifies a referent from the *universe of discourse*, which consists of the real-world entities that we know and talk about (see Haegeman (1994)). For instance, among the four noun phrases *husband*, *King George VI*, *the King*, and *his* that are used to refer to the discourse referent *King George VI* in our example, only the NP *King George VI* selects from the universe of discourse and therefore refers independently. Each of the remaining noun phrases depends on *King George VI* for its interpretation. For example, the NP

[1] In other words, the coreference relation is symmetric, reflexive, and transitive.

the husband only identifies a subset of the referents in the universe that satisfy the property of being a husband but does not select a particular referent from the universe. Although anaphora and coreference are similar in various respects, there are a number of subtle differences between the two linguistic phenomena.[2]

- Anaphora refers to the linguistic phenomenon of having a noun phrase refer to a previously mentioned entity in a text for its semantic interpretation and is therefore a *context-sensitive* relation. In other words, a pair of NPs <NP$_i$, NP$_j$> constitutes an anaphoric relationship if $i < j$ and NP$_j$ depends on NP$_i$ for its interpretation, where NP$_k$ denotes the kth NP in a document. For instance, the NP pair <*Queen Elizabeth*, *her*> forms an anaphoric relationship in our example. On the other hand, coreference refers to the problem of identifying all noun phrases that refer to the same entity in a text and is not necessarily context-sensitive. For example, the noun phrases *Queen Elizabeth* and *the Queen Mother* can both be understood to refer to the same real-world entity in the absence of context and by definition they form a coreference (but not anaphoric) relationship.

- Anaphora, unlike coreference, is not an equivalence relation. Since an anaphoric entity cannot depend on itself for its semantic interpretation, the anaphoric relation is not reflexive. In addition, the relation is by definition an order-dependent relation and is therefore asymmetric with respect to the two NPs involved. As we have seen before, the pair <*Queen Elizabeth*, *her*> constitutes an anaphoric relationship, but <*her*, *Queen Elizabeth*> does not.[3] Transitivity is undefined for the anaphoric relation, since an NP either refers independently or depends on other NPs for its interpretation.

- Some anaphoric relations are not coreferential. In bound anaphora, the anaphor is not coreferential with the antecedent, as exemplified in the sentence *Each of Diana's former police bodyguards had his opinion about the late princess' private life*, where the pronoun *his* lies within the scope of the universal quantifier *each* and refers to each of the persons that is being quantified over. In other words, *his* is not considered coreferent with *each of Diana's former police bodyguards*.

- Some coreferential relations are not anaphoric. Anaphora is a within-document phenomenon, whereas cross-document coreference is possible. For example, two occurrences of *Queen Elizabeth* in different documents can be considered coreferent, whereas a pronoun *her* in document x cannot form an anaphoric relationship with *Queen Elizabeth* in document y even if the pronoun refers to *Queen Elizabeth* in document x.

For the most part, however, we will not attempt to make a distinction between anaphora and coreference in this paper. In fact, we will restrict our attention to within-document entity coreference, viewing coreference resolution essentially as anaphora resolution, in which we attempt to identify an antecedent for each "anaphoric" noun phrase by searching for a preceding noun phrase for each discourse entity that is potentially coreferent with it. This implies that *any* noun phrase that is part of a coreference chain but is not the head of the chain will be considered "anaphoric" under our definition. Linguistically speaking, however, this definition is by no means a self-imposed limitation on the scope of coreference resolution, as the goal of partitioning a set of

[2]See Hirst (1981) and van Deemter and Kibble (2000) for a more detailed description of the differences between anaphora and coreference.

[3]As a side note, forward references in which an NP depends on a referential expression that appears after it in the associated text for its interpretation are known as *cataphoric* NPs and are not considered anaphoric.

discourse entities into coreference equivalence classes remains the same. As we will see, treating all types of noun phrases as potentially anaphoric enables us to develop a conceptually simple solution to the problem and is therefore an advantage from a computational perspective.

1.2 Goals

At a high level, we propose the investigation of computational approaches to coreference resolution as the primary area of study for the dissertation. Specifically, our goal is to investigate whether it is possible to improve existing machine learning approaches to coreference resolution. Our decision to focus on machine learning approaches is motivated by the fact that machine learning approaches have been applied reasonably successfully to this problem, and have been shown to be a promising way to build robust and portable coreference resolution systems. In support of our proposed research, we will address the following in this paper:

- We account for the inherent difficulty of coreference resolution and discuss the practical need for a solution to the problem.
- We discuss existing algorithmic solutions to the problem by reviewing previous work on anaphora and coreference resolution. In particular, we focus on computational approaches to the problem, discussing recent research efforts that involve the development of algorithms for anaphora and coreference resolution. (See Botley and McEnery (2000) for an overview of linguistic and cognitive approaches to discourse anaphora.)
- We point out a number of issues in existing machine learning approaches that are either not explored or being overlooked and can potentially lead to a deterioration in system performance. We examine each of these issues in detail and suggest potential solutions.
- We explain how far we have gotten to achieve our goal and what remains to be done.

The remainder of the paper is organized as follows. Section 2 provides an empirical justification of the hardness of coreference resolution and discusses the potential applications of the problem. Section 3 describes a generic algorithm that can be regarded as a skeleton behind the design and development of most contemporary anaphora and coreference resolution algorithms. In section 4, we compare and contrast existing approaches to anaphora and coreference resolution in the context of the generic algorithm presented in section 3. Section 5 discusses the goals and contributions of our research in more detail, describing what we have done and what remains to be done in order to achieve our goal.

2 Coreference Resolution as a Hard and Important Problem

As noted above, this section discusses the difficulty of coreference resolution as well as the practical aspects of the problem. Coreference resolution is an interesting task by itself, partially owing to the inherent difficulty of the problem. While anaphora resolution has drawn considerable interest in the computational linguistics community for more than three decades, the problem is far from being solved. In fact, anaphora/coreference resolution has widely been considered to be one of the most challenging problems in AI. There has been prevailing consensus that the difficulty of the problem lies in its dependence on sophisticated semantic and world knowledge. Here we will elaborate further on this by providing some justifications with respect to coreference resolution, but the arguments carry over to anaphora resolution as well.

- **Many sources of information play a role.** For instance, lexical information such as head noun matches (as in *Lord Spencer* and *Spencer*) is an indicator of coreference, although it is not an absolute indicator (e.g. *Lord Spencer* and *Diana Spencer* are not coreferent). In addition, syntactic constraints such as the binding constraints place restrictions on the interpretation of different kinds of noun phrases. Furthermore, knowledge sources such as gender and number, semantic class, discourse focus, and world knowledge also play a role in determining whether two discourse entities are coreferent.

- **No single source of knowledge is completely reliable indicator.** For example, two semantically compatible NPs are potentially coreferent (e.g. *Diana Spencer* and *the princess*), but whether the NPs are actually coreferent depends on other factors (such as contextual information). Additional complications of the problem come from the fact that even traditional linguistic constraints indicating (non-)coreference such as number (dis)agreement is not absolutely hard (e.g. the singular NP *assassination (of her bodyguards)* can be coreferent with the plural NP *these murders*).

- **Coreference strategies differ depending on the type of NP.** For example, definite NPs are more likely to be anaphoric than their non-definite counterparts (e.g. the article immediately preceding *man* in the sentence *Diana saw the/a photographer following her secretly* determines whether the NP has an existential or definite reading). Furthermore, pronoun resolution, which can be viewed as coreference resolution applicable only to pronouns, is notoriously difficult, partly because resolution strategies differ for each type of pronouns (e.g. reflexives versus possessives) and partly because some pronouns such as pleonastic pronouns are semantically empty and therefore need to be handled separately (e.g. the pronoun *it* in the sentence *Camilla went outside and it was raining* is pleonastic).

Since the problem relies on knowledge at all levels of language processing, including lexical, syntactic, semantic, and world knowledge, success in this problem is likely to be indicative of advancement of language technologies. If we adopt the view that anaphora resolution is simply coreference resolution applied to a subset of the NPs that the latter problem normally considers, then we can argue that coreference resolution is at least as hard and potentially as interesting as anaphora resolution.

Additionally, the role of coreference resolution as a key task in many NLP applications also makes it an important problem in language understanding. For illustrative purposes, we here enumerate several applications can potentially benefit or have actually benefited from the availability of coreference information.

Question Answering. In its most general form, the task of question answering is concerned with finding an answer to a natural language question from a large text collection or determining that an answer cannot be found. Question answering systems are useful in that they provide the exact information users need and reduce their burden of having to search through the documents returned by search engines in response to their queries. To see the connection between question answering and coreference resolution, consider the query *Where was Mozart born?*. A question answering system may first retrieve the sentence *He was born in Salzberg* from a document talking about *Mozart*. In this case, the system will return the correct answer if it can determine that the pronoun

He is coreferent with *Mozart*. In fact, several question answering systems have begun to make use of coreference information and shown improvement in system performance (Breck et al. (1999); Singhal et al. (1999); Morton (2000)). In a question answering system, each sentence in a document that is retrieved for a given question is usually assigned a numeric score based on the sum of the term weight of each overlapping word between the question and the sentence (Singhal et al., 1999), and sentences with the highest scores are assumed to contain potential answers. Morton (2000) augments the above algorithm by considering all the terms that are in the same coreference chain as any of the noun phrases in a sentence to have appeared in that sentence as well, giving these additional terms 90% of the normal term weight. By altering the ranking of each potential answer based on identified coreference relations, Morton reports improvement in the performance in his coreference system.

Text Summarization. Summarization helps users focus on or keep track of the "important" information contained in a large pool of documents, either by providing a short summary of a document (single-document summarization) or by synthesizing information about the same event mentioned in different documents (multi-document summarization). Because of the usefulness of the task, the newest version of Microsoft Word is capable of text summarization, for example. Although most text summarization systems have not yet taken into account coreference information, there is initial evidence that it would be helpful to do so. For example, Baldwin and Morton (1998) describe a method for generating extraction-based, query-dependent summaries. Not only do they include in the summary those sentences that contain a term appearing in the query, they also incorporate sentences containing a noun phrase that is coreferent with a term occurring in a sentence already selected by the system. On the other hand, Azzam et al. (1999) describe a heuristic-based method for selecting the "best" coreference chains to represent the main topic of a text.

Cross-Document Entity Coreference. The goal of cross-document entity coreference is to determine whether a given pair of noun phrases occurring in different documents refers to the same entity. Cross-document coreference is particularly useful for text summarization systems that need to identify and merge the same piece of information about an entity mentioned in different documents in order to avoid repetition. Bagga and Baldwin (1998) show how within-document coreference information can potentially be used to perform cross-document coreference. Specifically, for any two NPs, NP_i from document 1 and NP_j from document 2, that are potentially cross-document coreferent, the cross-document coreference system first extracts all and only those sentences having a noun phrase that is coreferent with NP_i in document 1. Another set of sentences is similarly extracted from document 2 using NP_j. The vector space model (Salton et al., 1975) is then applied to compute the similarity between these two "documents". If the similarity score is above a certain threshold, then NP_i and NP_j are considered to be cross-document coreferent.

Information Extraction. An information extraction (IE) system takes as input a text relevant to a given domain and automatically extracts specific pieces of information from the text. A number of Internet applications crucially rely on IE technologies, including NLP systems that construct knowledge bases (Craven et al., 1997) and weather forecast databases (Soderland, 1997) from Web pages. The importance of coreference resolution in IE is reflected in the decision to make coreference one of the four evaluated IE sub-tasks in MUC-6 (1995). Specifically, coreference information is used to merge discourse entities in a text, allowing the fusion of extracted information regarding the same entity from different sentences. This information in turn aids *template generation*, which determines the number of events described in the text and maps each piece of information being extracted onto the appropriate event (Cardie, 1997).

Machine Translation. There is an obvious need for high-quality machine translation (MT) systems. In most existing MT systems, an anaphora in the source language is simply be translated to the corresponding anaphora in the target language. Anaphora resolution comes into play when discrepancies exist between the two languages with respect to anaphor selection. For example, a pronoun in the Malay language is often translated directly by its antecedent (Mitkov, 1999). Anaphora resolution would also be useful in discourse translation, in which it is important to resolve an anaphoric entity to its antecedent.

3 A Generic Algorithm for Anaphora and Coreference Resolution

In this section, we will introduce a generic algorithm, GENERIC-RESOLVE for anaphora and coreference resolution. As noted in section 1, virtually all anaphora resolution algorithms and coreference resolution algorithms that view coreference as anaphora resolution are instantiations of GENERIC-RESOLVE. As we will see in subsequent sections, this algorithm provides a unified framework for characterizing and analyzing existing resolution algorithms. GENERIC-RESOLVE, shown in Figure 2, takes as input an unrestricted text D and searches for an antecedent for each potentially anaphoric discourse entity in D. The first three steps of the algorithm are performed at the document level, whereas the remaining steps are performed for each discourse entity in the document. A step by step explanation of the algorithm follows.

Step 1: Identification of Discourse Entities For pronoun resolution algorithms, the task here is simply to identify all of the pronouns in the associated text. For coreference resolution algorithms, the task is to identify all of the noun phrases in the text.

Step 2: Characterization of Discourse Entities This step is composed of two sub-tasks. The first task is to define a representation of a discourse entity, or equivalently, a way to characterize a discourse entity. The representation precisely determines the knowledge regarding a discourse entity that the algorithm needs in order to perform the remaining steps successfully. Once a representation is created, the second task is then to compute the information specified in the representation for each entity.

Step 3: Anaphoricity Determination The goal of ANAPHORICITY DETERMINATION is to determine whether a discourse entity is anaphoric or not. Non-anaphoric entities, by definition, do not possess an antecedent and hence the algorithm will not need to search for an antecedent for these entities if anaphoricity information is available. Some algorithms do not perform anaphoricity determination, in which case all discourse entities are implicitly assumed to be potentially anaphoric.

Step 4: Generation of Candidate Antecedents Unlike the previous steps, this step and the remaining ones are performed for each anaphoric entity in the document. Once an NP is determined to be anaphoric, the algorithm identifies the scope of the NP and gathers those NPs that fall within its scope to generate the list of candidate antecedents for it. For most anaphora and coreference resolution algorithms, the scope of the anaphor is not identified for computational reasons such as

GENERIC-RESOLVE

Input: An unannotated document D

Algorithm:

/* 1. IDENTIFICATION of discourse entities from D */
Produce a list of discourse entities $E := \{ e \mid e$ is a discourse entity that appears in $D \}$.
Initialize the list of potentially anaphoric entities $A := E$.

/* 2. CHARACTERIZATION of discourse entities */
Compute for each discourse entity NP$_i$ a set of values $\{ k_{i1}, k_{i2}, \ldots, k_{im} \}$ from m knowledge sources.

/* 3. ANAPHORICITY DETERMINATION (**Optional**) */
Compute the list $A := A \setminus \{ e \in A \mid e$ is not anaphoric $\}$.

foreach NP$_j$ in A **do**

/* 4. GENERATION of candidate antecedents */
Compute the list of candidate antecedents of NP$_j$ $C_j := \{ e \in E \mid e$ lies within the scope of NP$_j$ $\}$.

/* 5. FILTERING (**Optional**) */
Compute the list $C_j := C_j \setminus \{ e \in C_j \mid e$ satisfies a non-coreferent constraint with NP$_j$ $\}$.

/* 6. SCORING/RANKING (**Optional**) */
Score or rank each NP in C_j and sort C_j w.r.t. the score.

/* 7. SEARCHING/CLUSTERING */
Select an antecedent for NP$_j$ from C_j.

endfor

Output: A document D annotated with anaphoric information

Figure 2: The GENERIC-RESOLVE Algorithm. In this algorithm, a list is viewed as a set whose elements can be ordered and hence all basic set operations can be applied.

complexity or accuracy, and the list of candidate antecedents in this case is simply taken to be the list of NPs preceding the anaphor in the associated document.

Step 5: Filtering FILTERING involves removing unreasonable candidate antecedents for each anaphoric noun phrase. Specifically, FILTERING is performed based on a set of rules or hard constraints that specify non-coreference or contra-indexing between two discourse entities. A potential antecedent that satisfies any of the non-coreferent constraints can therefore be removed from the candidate list. This step aims at reducing the amount of processing that needs to be performed by the algorithm, and can potentially improve the accuracy of the resolution procedure.

Step 6: Scoring/Ranking SCORING, or RANKING, of the remaining candidates is performed based on a set of rules or soft constraints that indicate preferences that two NPs co-specify. In some cases, each candidate antecedent for a given anaphor is assigned a numeric score that reflects the likelihood that the two NPs under consideration possess an anaphoric or coreferent relationship, and the list of antecedents is subsequently sorted based on the scores. In other cases, only a ranking of the candidate antecedents is generated based on a set of rules or discourse principles. Overall, the goal of this step is to rank candidate antecedents according to the preference adopted by a specific algorithm.

Step 7: Searching/Clustering The goal of this step is to select an antecedent for a given anaphor from the list of candidate antecedents returned by the previous step. If this list is empty, then no antecedent will be selected for the anaphor. If SCORING or RANKING of the candidate antecedents has been performed by step 6, which is optional, then SEARCHING becomes the trivial task of selecting the first (or highest-ranking) element in the candidate list as the antecedent of the anaphor. Otherwise, this list of candidate antecedents is searched in some order specified by the resolution algorithm, and the "best" NP encountered is then selected as the antecedent. In the case of coreference resolution, this process is essentially equivalent to applying a single-link clustering algorithm to each anaphoric NP to cluster the NPs in the document and generate a partition on them.

Note, however, that steps 3, 5 and 6 can be absent in an anaphora or coreference resolution algorithm, although it is not common to have an algorithm in which all three steps are omitted. Existing resolution algorithms differ in the way these seven steps are implemented. As a result, the generic algorithm can be thought of as providing a natural way of characterizing resolution algorithms. We will discuss this issue in detail in the next section.

4 Related Work

As we mentioned before, we will attempt to characterize existing computational approaches to anaphora and coreference resolution in the context of the GENERIC-RESOLVE algorithm in this section. We will begin with a discussion of several trends that are representative of ongoing research in this area. As each trend normally signifies a shift in the overall research direction, the discussion here will set the stage for comparison between different approaches in the subsequent subsections.

4.1 Research Trends

Research on anaphora and coreference resolution in computational linguistics has proceeded in various directions ever since its inception three decades ago. Nevertheless, we are able to observe the following research trends with respect to this problem.

From Knowledge-Rich Approaches to Knowledge-Lean Approaches. Anaphora and coreference resolution systems differ with respect to the *types* of knowledge source required to perform accurate resolution. This difference separates *knowledge-rich* systems — which usually require domain-specific knowledge, semantic and discourse analysis, as well as sophisticated inference mechanisms — from *knowledge-lean* systems, which deliberately avoid the use of sophisticated knowledge and complex analyses in the resolution procedure and instead rely only on morphological and possibly (shallow) syntactic information. While early research on this problem focuses on knowledge-rich approaches, the need for fast and robust solutions to anaphora and coreference resolution has prompted the development of knowledge-lean approaches (Mitkov et al., 2001).

From Semi-Automatic Pre-Processing to Fully Automatic Pre-Processing. An anaphora or coreference resolution system takes a raw text, not an annotated text, as input. Consequently, an input text must be pre-processed before any given resolution procedure can be applied to it. The first two steps in GENERIC-RESOLVE essentially outline the common pre-processing steps: IDENTIFICATION and CHARACTERIZATION of discourse entities. Specifically, a coreference system needs to identify all noun phrases from an input text. During CHARACTERIZATION, knowledge required by the resolution algorithm is computed for each discourse entity identified in the first step. For instance, *gender* and *number* are two common knowledge sources employed by resolution systems. Neither NP identification nor knowledge computation can be achieved with perfect accuracy, however. Early research in anaphora and coreference resolution usually assumes that pre-processing is error-free. This entails manual fixing of errors made by the pre-processing routines. On the other hand, the focus of recent research has shifted towards the construction of fully automatic anaphora and coreference resolution systems in which the entire process is completely free from human intervention.

From Small-Scale Evaluation to Large-Scale Evaluation. Evaluation of early anaphora resolution algorithms is usually performed by hand. In fact, some of these algorithms were not implemented as computer programs at the time of evaluation. Since the evaluation process was not automatic, these algorithms were mostly evaluated on a relatively small number of unseen cases. The recent availability of a huge number of online texts as well as the development of fairly robust pre-processing routines such as morphological analyzers and shallow parsers have aided the automation of anaphora resolution systems and made automatic, large-scale evaluation of resolution algorithms possible.

From Knowledge-Based Approaches to Corpus-Based Approaches. Early anaphora resolution systems adopt knowledge-based approaches, in which the resolution procedure is based on a set of hand-crafted heuristics that specify whether two discourse entities can or cannot have an anaphoric relationship. Owing to the availability of corpora annotated with coreference information and the success of data-driven approaches in many NLP tasks, researchers in recent years have attempted to apply corpus-based/empirical methods to anaphora and coreference resolution, in which anaphora or coreference knowledge is derived automatically from labeled data.

In principle, we can characterize existing work on anaphora and coreference resolution along any of the above dimensions. Nevertheless, we will mainly focus our discussion on the last dimension, knowledge-based versus corpus-based approaches, in sections 4.2 and 4.3 for the following reasons.

First, as we will be exploring issues involving the machine learning of coreference resolution in section 5, a characterization of related work along the last dimension seems most pertinent to the subsequent discussion. More importantly, the trend along the other dimensions typically follows from the last dimension — corpus-based methods allow large-scale evaluations of coreference systems that employ fully automatic pre-processing and knowledge-lean methods. Alternatively, the other trends seem to go together. In most cases, for example, large-scale evaluations are possible only if fully automatic pre-processing is employed. In spite of the relationship that exists among the dimensions, we will briefly discuss the remaining three dimensions in sections 4.4, 4.5 and 4.6.

4.2 Knowledge-Based Approaches

As we mentioned before, virtually all anaphora and coreference resolution algorithms are instantiations of the generic algorithm described in section 3. As anaphora and coreference resolution systems adopting knowledge-based approaches differ from those adopting corpus-based methods primarily with respect to the way the FILTERING step and the SCORING step are implemented, the comparison between the various approaches will be based mainly on these two steps. In particular, IDENTIFICATION of discourse entities is trivial for pronoun resolution and is performed similarly among coreference resolution systems using a noun phrase finder. In addition, GENERATION of candidate antecedents for a potential anaphor in most systems simply involves the construction of a set of noun phrases preceding the anaphor under consideration in the associated document, although some systems impose a limit on how far apart an anaphor and a candidate antecedent can be. Consequently, these two steps will not be discussed explicitly, although other steps will be mentioned as necessary. In the rest of this subsection and the next, we will discuss related work using knowledge-based approaches and corpus-based approaches respectively.[4]

For systems adopting knowledge-based approaches, the rulesets used in FILTERING and SCORING are composed entirely of hand-coded constraints. As noted above, FILTERING involves removing any candidate antecedent that violates any of the non-coreferent constraints from the candidate list. Some commonly pre-specified constraints employed by knowledge-based systems are traditional linguistic constraints such as agreement constraints. In addition, the score associated with each soft constraint used in the SCORING step of knowledge-based systems is also pre-specified rather than learned. Broadly speaking, knowledge-based systems can further be divided into three classes: syntax-based approaches, discourse-based approaches, and factor-based approaches.

4.2.1 Syntax-Based Approaches

Syntax-based anaphora resolution algorithms rely solely on syntactic and morphological information and assume the existence of a full parse tree for each sentence in the input text. Both FILTERING and RANKING occur during the SEARCHING step. Specifically, for each potential anaphor, the search for an antecedent is performed via the traversal of parse trees. When a candidate antecedent is encountered and it does not survive the FILTERING step, the search for the next candidate continues. Otherwise, the candidate is selected as the antecedent. RANKING of candidate antecedents is encoded implicitly in the order in which the tree nodes (and hence the candidates) are visited.

Hobbs' naive algorithm (1978) is one of the most well-known syntax-based algorithms for resolving pronoun references (Tetreault, 1999). Specifically, the algorithm considers the sentences in the text in reverse order, starting from the sentence in which the pronoun resides and searching

[4]A more detailed description and complementary analysis of most of the systems discussed below can be found in Mitkov (1999).

for potential antecedents in the corresponding parse trees in a left-to-right, breadth-first manner that obey binding and agreement constraints. The algorithm's preferences for recency (i.e. NPs that are in closer proximity to the anaphor) as well as for NPs in the subject position are generally believed to be the reason for its good performance on pronouns with intra-sentential antecedents (Lappin and Leass, 1994, Walker, 1989). Despite its simplicity, the algorithm has received criticisms regarding its dependence on full parse trees (Ge et al., 1998).

4.2.2 Discourse-Based Approaches

Discourse-based methods for anaphora resolution have been developed out of the belief that some types of anaphora can only be resolved with an understanding of the focus (or center) of the discourse. For instance, consider the following sentence.

`William saw Harry in the park today.  He was talking to his friend.`

Evidently, the resolution of the pronoun *He* cannot be based solely on grammatical information such as gender and number, since both *William* and *Harry* are potential antecedents of *He*. Nevertheless, there is an indication that the center of the discourse has shifted from *William* to *Harry* in the first sentence. Consequently, most readers would perceive *He* as referring to *Harry* and not *William*, since pronominalization serves to capture attention and hence the noun phrase that is the focus at a particular point in the discourse is more likely to be pronominalized than the other NPs.

Discourse-based anaphora resolution algorithms rely on principles such as coherence and focusing to track the center of a discourse. Discourse-level constraints and preferences formulated from discourse principles are then incorporated into the FILTERING step and the RANKING step of the resolution procedure respectively, in addition to syntactic and morphological constraints that are commonly applied in these steps by other types of anaphora resolution algorithms. Algorithms in this category differ from those in other categories primarily with respect to the way RANKING is performed: for discourse-based algorithms, ranking of candidate antecedents is driven predominantly by discourse principles, as opposed to syntax-based methods which do not perform any ranking and factor-based methods which score candidate antecedents based on a number of factors. Below we subcategorize discourse-based resolution algorithms. Section 4.2.2.1 introduces algorithms that are built upon centering theory, and section 4.2.2.2 discusses algorithms that are based on a more general notion of focus.

4.2.2.1 Centering Theory and Centering Algorithms

Grosz and Sidner (1986) develop a theory of discourse structure, and the part of the theory that has sparked a lot of research on focus-based anaphora resolution algorithms is *centering* (Grosz et al., 1995). Centering theory models the local coherence of a discourse, and is composed of a set of constraints governing center movement (the conditions under which the center of a discourse can or should move from one discourse entity to another) and center realization (the conditions under which a discourse entity can be pronominalized) that are formulated based on psycholinguistic and cross-linguistic evidence. Generally speaking, there are two principles underlying centering theory: cohesion (maintaining the same center) and salience (realizing the center as the most prominent NP). Note, however, that centering is a theory for interpreting pronouns in a discourse and its use in anaphora and coreference resolution is only one of the applications of the theory. Anaphora resolution algorithms that are built upon centering constraints are collectively known as *centering algorithms*.

The BFP centering algorithm. One of the most well-studied centering algorithms is the BFP algorithm (Brennan et al., 1987). BFP can also be viewed as an instantiation of GENERIC-RESOLVE. Specifically, FILTERING of candidate antecedents for a given anaphor in BFP makes use of syntactic constraints such as the binding constraints and morphological constraints such as gender and number. The RANKING of candidate antecedents, however, is guided by centering principles.

Improvements to BFP. A series of studies (Kehler, 1997a, Walker, 1989, Tetreault, 1999) reveals several weaknesses of the BFP algorithm, some of which also exhibited by other centering algorithms. First, the algorithm is weak in resolving pronouns that refer to a global focus, since centering only captures local discourse phenomena. Second, being built upon a cognitive model, BFP does not allow for the incremental processing of pronouns occurring in the same sentence. Third, BFP does not have an intrasentential processing mechanism, since the algorithm attempts to search for antecedents for a given pronoun only in the previous utterances. Finally, the ranking of candidate antecedents is not completely specified by the algorithm, which means that it is possible for two candidates to have the same ranking.

There have been various proposals that attempt to remedy the above problems. For instance, Walker (1989) describes a way of incorporating a global focus in the centering framework. Tetreault (1999) proposes a new centering algorithm, Left-Right Centering (LRC), that can be viewed as extending the Hobbs algorithm with discourse information to direct the search for antecedents and therefore can resolve pronouns incrementally. Furthermore, Kameyama describes a method, which she calls *intrasentential centering*, to enhance BFP with an intrasentential processing mechanism (Kameyama, 1998). Finally, if potential antecedents for an anaphor can be proposed one after the other as in the Hobbs algorithm, there would be no ambiguity of preference for antecedents.

4.2.2.2 Focus-Based Approaches

The role of focus in anaphora resolution was recognized even before centering theory was developed. Unlike centering algorithms, some of the focus-based algorithms described here are designed to resolve definite noun phrases rather than pronouns.

Grosz's stack-based approach. Grosz (1977), for instance, defines a representation, *focus space*, for tracking the focus of a discourse in task-oriented dialogs. More precisely, each focus space comprises all entities in the same discourse segment. An *explicit focus space* contains all discourse entities explicitly mentioned in the dialog, whereas an *implicit focus space* contains all entities that are either related to explicitly mentioned entities or implicitly referred to in the dialog. The current discourse segment, or the *active focus space*, is on the top of the focus stack, and a list of *open focus spaces* that consist of entities that might be referred to later in the dialog is kept in the stack. The active focus space is popped off the stack when there is a focus shift. The discourse entities in each focus space are represented in an elaborate semantic network. When attempting to resolve a definite NP during SEARCHING, the search procedure is constrained by considering a "focused match" only between the network in which the definite NP resides and the network representing the active focus space. Consequently, the candidate list is composed exactly of the entities in the active focus space, and NPs residing in other focus spaces are effectively filtered from the list. There are two problems associated with Grosz's approach, however. First, the approach can only handle definite descriptions but not pronominal references. In addition, the approach seems to be applicable only to task-oriented dialogs. It is unclear how the approach can be extended to other other genres where it may be harder to detect a shift in focus.

Sidner's mutual bootstrapping method for anaphora resolution. Sidner (1979, 1981) develops a local focusing framework that extends Grosz's approach to enable the resolution of pronominal references and recognize focus shifts in broader situations. She argues that the two processes, *focus tracking* and *anaphora resolution*, influence each other, and her framework comprises two algorithms, the *focusing* algorithm and the *anaphora resolution* algorithm, that reflect this circular dependency. To represent the current state of a discourse, Sidner's framework maintains a set of data structures, including the *current focus*, the *actor focus* that is used to handle agentive pronouns specifically, a list of *alternative candidate foci*, and a *focus stack*. For each sentence, the focusing algorithm uses a set of rules as well as results from anaphor interpretation to determine whether there is a shift in focus and updates the data structures accordingly. Then, for each anaphor encountered, the anaphora resolution algorithm uses another set of rules to rank candidate antecedents based on the focus-tracking data structures. Since the algorithm only considers NPs in these data structures as candidate antecedents, implicit FILTERING of discourse entities occurs when the focusing algorithm updates the data structures.

Azzam's extension to Sidner's algorithm. Azzam (1996) points out two problems with Sidner's approach. First, Sidner's algorithm is designed to handle sentences with relatively simple structures, i.e. sentences of the form *Subject Verb Object* optionally followed by prepositional phrases and adverbial adjuncts. A complex sentence must be segmented into simpler constituents before her algorithm can be applied, but Sidner does not provide any mechanism for doing so. Second, the focus-tracking data structures do not contain entities from the sentence under consideration. Since the anaphora resolution algorithm proposes only candidate antecedents residing in these data structures, intrasentential antecedents are virtually impossible. Azzam thus proposes a simple extension to Sidner's framework that handles both problems simultaneously : making embedded sentences (i.e. sentences that correspond to elementary events) the unit of discourse processing. She then presents an algorithm for extracting embedded sentences from complex sentences and applies Sidner's algorithm to embedded sentences instead. For evaluation, she extends the coreference resolution algorithm used in the LaSIE information extraction system (Gaizauskas et al., 1995) with her focusing mechanism, but concludes that there is no observable difference between the performance of the coreference algorithms with and without focusing (Azzam et al., 1998).

Strube's S-list algorithm. Strube (1998) proposes a new pronoun resolution algorithm that can be viewed as an alternative to centering. Specifically, a simple data structure, S-list, is used to store all discourse entities in the previous and current utterances, effectively FILTERING or eliminating all NPs that do not exist in the previous or current utterance from consideration. SCORING occurs immediately after each discourse entity encountered is added to the list, which is sorted such that its elements are always in decreasing salience with respect to the current discourse. The way salience is computed roughly corresponds to Prince's *familiarity scale* (Prince, 1981). SEARCHING proceeds simply by selecting the first element in the list that satisfies the agreement constraints with the pronoun.

4.2.3 Factor-Based Approaches

Factor-based anaphora resolution algorithms seek to combine various knowledge sources, including morphological, syntactic, semantic, and in some cases pragmatic information, for antecedent selection. These "factors" can be broadly divided into two categories: constraints (that must be satisfied) and preferences (that distinguish between candidates satisfying all constraints). In contrast to discourse-based methods, factor-based methods do not rely on an elaborate discourse theory that usually involves complex modeling of attentional states, and antecedent selection is not predominated by discourse principles, which have an intrinsic preference for intersentential antecedents (Walker, 1989). In particular, discourse information in this class of algorithms is formulated as preferences rather than constraints.

When viewed as instantiations of GENERIC-RESOLVE, factor-based anaphora resolution algorithms operate as follows: To resolve a potentially anaphoric noun phrase, constraints are first applied to narrow down the set of potential antecedents during the FILTERING step. Subsequently, each preference factor (independently) proposes an antecedent for the noun phrase under consideration during the RANKING/SCORING step. Strictly speaking, each preference factor represents a constraint that can either be hard (in which case only one antecedent is selected) or soft (in which case a numeric score is assigned to each candidate antecedent reflecting the preference for antecedence). The possibly contradictory individual decisions are then coordinated and combined into a global decision via the use of a predefined metric. The SEARCHING step then relies on the global decision reached for each candidate antecedent to select the antecedent for the anaphoric noun phrase. Despite these similarities, algorithms in this class differ with respect to the *linguistic knowledge sources* being used as well as the *strategies* being employed to combine these knowledge sources. Although existing work on anaphora and coreference resolution has generally focused on investigating the knowledge sources that are useful for the task, Mitkov (1997) shows that extra-linguistic strategies (e.g. a traditional approach in which "bad" candidate antecedents are filtered before they are ranked versus an uncertainty-based approach in which *all* potential antecedents (i.e. no filtering) are assigned a score based on each available knowledge source) also play an important role in the performance of an anaphora resolution algorithm. As a result, we will discuss this class of resolution algorithms in terms of both the knowledge sources and the strategies used to combine them in the rest of the subsection.

Carbonell and Brown's multi-strategy approach. Carbonell and Brown (1988) develop an anaphora resolution algorithm under the hypothesis that anaphora resolution can be best accomplished via "a combination of a set of strategies rather than by a single monolithic method". Various constraints for anaphora resolution are proposed: syntactic (e.g. gender and number), semantic (e.g. selectional constraints), and pragmatic (e.g. eliminate a candidate antecedent from consideration if some action that occurs between the antecedent and the anaphor implies that the two cannot possess an anaphoric relationship). As far as preferences are concerned, factors such as recency, topicalization (i.e. entities that appear in topicalized structures), syntactic parallelism (i.e. entities having the same grammatical role as the anaphor) and semantic parallelism (i.e. entities having the same thematic role as the anaphor) are employed for antecedent selection. Knowledge sources are combined to select an antecedent for an anaphor as follows: candidate antecedents that violate any of the constraints are filtered out. If more than one candidate is left, the preferences are employed. If more than one preference applies and each applicable preference proposes a different antecedent, then the anaphor is considered to have a truly ambiguous antecedent.

Lappin and Leass's syntax-based approach. Lappin and Leass (1994) propose a pronoun resolution algorithm that relies on a set of syntax-based constraints and salience-based preferences. In contrast to Carbonell and Brown (1988) who make use of semantic and pragmatic constraints that are generally hard to encode with a reasonable accuracy, Lappin and Leass instead employ only morphological constraints such as gender and number and syntactic constraints such as the binding constraints but assume perfect output from a morphological analyzer and a full syntactic parser.

Additionally, as opposed to most previous work in which pleonastic pronouns are manually filtered out during evaluation, Lappin and Leass design a separate set of patterns to identify pleonastic pronouns explicitly. During SCORING, an anaphoric entity is resolved to the most salient preceding entity, where the salience of a noun phrase is given by the salience of the coreference equivalence class to which the noun phrase belongs. The salience of a class is in turn calculated based on a set of *salient factors* applied to each member of the class. Each salience factor is associated with an initial weight that indicates the relative contribution of the factor to overall salience. As these initial weights are degraded after each utterance in the discourse is processed, the salience of each equivalence class constructed so far needs to be re-calculated accordingly. The salience factors are fairly intuitive. For example, both sentence recency and the grammatical role of a discourse entity play a role in determining the salience.

Kennedy and Boguraev's parser-free pronoun resolution system. One advantage of Lappin and Leass's algorithm is that it does not depend on semantic or pragmatic information for pronoun resolution. However, the availability of perfect parse trees is too strong an assumption to make in realistic settings. To further relax this assumption, Kennedy and Boguraev (1996) propose an algorithm that is structurally equivalent to that of Lappin and Leass except that the algorithm does not rely on deep syntactic parsing. Specifically, the information provided by the full parser in Lappin and Leass's algorithm (such as the grammatical role of an NP) is approximated by a set of ten heuristics in Kennedy and Boguraev's algorithm.

Baldwin's high-precision pronoun resolution system. Baldwin (1997) describes a high-precision rule-based pronoun resolution algorithm, CogNIAC. Like Kennedy and Boguraev's algorithm, CogNIAC only has access to morphological information and shallow syntactic information that is heuristically computed. The algorithm is composed of a set of rules that aims to recognize anaphoric relationships. The rules are ordered according to precision and importance with respect to the pronoun resolution task and are successively applied to the pronoun under consideration. The resolution of a pronoun terminates as soon as a rule applies to the pronoun.

Mitkov's knowledge-lean robust pronoun resolution system. Mitkov (1998) acknowledges the importance of developing knowledge-lean approaches to anaphora resolution and proposes a robust algorithm for resolving pronouns in technical manuals. Like CogNIAC, Mitkov's algorithm relies solely on morphological and shallow syntactic information provided by a part-of-speech tagger and a base noun phrase finder respectively. During FILTERING, any potential antecedent that either violates agreement constraints or is more than two sentences apart from the pronoun under consideration is removed from the candidate list. Each remaining candidate is scored based on a set of antecedent indicators, each of which potentially affects the salience of an NP and effectively serves as a preference for antecedent selection. A positive score is assigned to a candidate by a boosting indicator and a negative score assigned by an impeding indicator. The candidate with the highest aggregate score is then proposed as the antecedent. While the algorithm has been applied only to technical manuals, Mitkov argues that the majority of the antecedent indicators appear to be genre-independent. Unlike most previous work in which the salience of an NP is dependent almost exclusively on positional information (e.g. recency) and grammatical role, some of Mitkov's antecedent indicators make use of lexical information and NP type as additional knowledge sources. For example, impeding indicators are applicable not only to NPs embedded in PPs but also to indefinite NPs, and boosting indicators favor lexically reiterated items in addition to NPs appearing in section headings. Finally, the genre-specific indicators that the algorithm uses include preferences for NPs that represent terms used in the genre as well as those that are objects of certain "indicating verbs".

Cardie and Wagstaff's clustering approach. Cardie and Wagstaff (1999) explicitly view coreference as a clustering task and use a right-to-left single-link clustering algorithm to coordinate the application of constraints and preferences for partitioning the given noun phrases into coreference equivalence classes. Specifically, they define a distance metric between two noun phrases that is essentially a linear combination of the (in)compatibility scores computed from the (relational) constraints and preferences. During CLUSTERING, if the distance between two noun phrases is less than the predefined clustering radius, then the corresponding equivalence classes are considered for possible merging. A wide variety of knowledge sources (encoded as features) are used by the clustering algorithm: 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). Like Mitkov's system, the weight associated with each knowledge source in Cardie and Wagstaff's distance metric is chosen by hand rather than determined empirically.

In all of the factor-based approaches (except Baldwin (1997)) discussed in this subsection, the relative importance of each knowledge source to overall performance is not evaluated. As we will see in the next section, one potential advantage of corpus-based approaches over knowledge-based approaches is that the information regarding the relative importance of the factors comes directly with the induction of certain types of models (e.g. decision trees).

4.3 Corpus-Based Approaches

In contrast to knowledge-based approaches, corpus-based approaches rely on rulesets that are composed of soft constraints derived from a corpus annotated with anaphora or coreference information in FILTERING and SCORING. Corpus-based approaches operate primarily by recasting the problems of anaphora and coreference resolution as a classification task. For anaphora resolution, a model that determines the probability that an NP is an antecedent of a given anaphor is learned from a corpus. For coreference resolution, however, a model that determines the probability that two NPs are coreferent is learned instead; in addition, a separate clustering mechanism is needed to coordinate the possibly contradictory pairwise classifications and construct a partition on the set of NPs with one cluster for each set of coreferent NPs during CLUSTERING. Training data are typically created by relying on anaphora or coreference information from the training documents. The feature vector essentially describes the two NPs involved and the context in which they occur, comprising relational features (i.e. features that test whether some property holds for the NP pair under consideration) and non-relational features (i.e. features that test some property of one of the NPs under consideration). An instance is labeled as *positive* if the two NPs possess an anaphoric or coreference relationship, and labeled as *negative* otherwise. Corpus-based approaches differ from each other in terms of how the probabilistic model is learned and can further be divided into three classes: manual approaches, machine learning approaches, and statistical approaches. We will elaborate on each of these approaches in the following subsections.

4.3.1 Manual Approaches

Coreference systems in this class constitute a special case of corpus-based approaches in which the probabilistic model (which in this case is composed of a set of coreference rules) is mined from the corpus entirely by hand. Consequently, manual approaches are not widely adopted, since a lot of human intervention is required to derive the desired coreference information.

Harabagiu et al.'s knowledge-minimalist approach. Harabagiu et al. (2001) present a knowledge-minimalist approach for mining "easy" coreference rules from an annotated corpus. The goal is to acquire a set of (positive) rules for determining when two NPs are coreferent. The approach is based on the following observations:

- For any anaphoric noun phrase in a coreference chain[5], a positive training instance can be formed from the noun phrase and each of its preceding noun phrases in the same chain.
- One antecedent is sufficient to resolve an anaphor.
- Some coreference relationships require more knowledge and hence are harder to resolve than the others.

The first observation implies that more than one positive instance can be created from each anaphoric NP. The second observation implies that in principle, given n anaphors in the training set, we only need n positive instances to acquire the rules. These two observations together imply that we have the freedom to select positive instances from which to induce coreference rules, since the number of available positive instances is at least as large as the minimum number of positive instances needed to derive the rules. The third observation implies that intelligent selection of "easy" positive instances can potentially minimize the knowledge needed to perform coreference resolution. The Harabagiu et al. algorithm first mines a set of rules for covering easy instances that are heuristically selected. The set of coreference rules is then transformed into a corresponding set of soft constraints by estimating the accuracy of each rule on the training data. To generate a partition on a given set of noun phrases from an unseen text, the algorithm relies on a local-search algorithm to search for the best possible partition. Specifically, the local-search algorithm starts with a random partition, makes local modifications to the partition, and picks the best partition at each iteration, where the score of a partition is computed from the pairwise coreference probabilities given by the soft constraints. The coreference system achieves a very high accuracy (approximately 90%) when evaluated on the MUC-6 coreference data set. Nevertheless, the system makes the strong assumption that all and only the NPs involved in coreference relationships are presented to the system for analysis. It is unclear whether the techniques would work well in the more realistic setting in which the set of NPs extracted by a shallow parser (which is presumably much larger than the set of NPs involved in non-singleton coreference classes) is analyzed by the coreference system instead.

4.3.2 Machine Learning 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 and can be characterized in terms of the *knowledge sources* being employed, the method of *training data creation*, as well as the *learning algorithm* and the *clustering algorithm* being chosen.

[5] As we mentioned in section 1.1, we assume throughout that a noun phrase is anaphoric if it is part of a coreference chain but is not the head of the chain.

Aone and Bennett's knowledge-rich anaphora resolution system for Japanese. Aone and Bennett (1995) describe a trainable system for classifying different types of anaphora occurring in Japanese texts about joint ventures. Specifically, the instance representation consists of 66 features, including lexical (e.g. part-of-speech), syntactic (e.g. grammatical role), semantic (e.g. semantic class), and positional (e.g. distance between the potential antecedent and the anaphor) features. Two different methods are used to create positive training instances: *transitive* (i.e. an instance is formed between an NP and each of its preceding NPs in the same anaphoric chain) and *non-transitive* (i.e. an instance is formed between an NP and its closest preceding NP in the same anaphoric chain). Negative instances are generated by pairing an NP with each preceding NP that does not have an anaphoric relationship with it. The system then uses the C4.5 decision tree induction system (Quinlan, 1993) to train an anaphora classifier that determines whether two NPs possess an anaphoric relationship. A best-first single-link clustering algorithm is then used to generate a partition on the set of NPs.[6] Results show that the learned anaphora classifier achieves superior performance than the manually designed resolver.

McCarthy and Lehnert's domain-specific coreference system. McCarthy and Lehnert (1995) describe RESOLVE, a coreference resolution system that is intended for use within an information extraction system. As a system that classifies coreferent phrases in the domain of joint ventures, 3 of the 8 features being employed are domain-specific. For example, there are features that test whether each of the NPs in the pair refers to a joint venture company (i.e. a company that is formed because of a tie-up among two or more entities). The domain-independent features can be characterized as lexical (e.g. whether the two NPs share a common noun phrase), semantic (e.g. whether one NP is an alias of the other), and positional (e.g. whether the two NPs are in the same sentence). No syntactic features are used. To generate positive training instances from coreference chains, only the *transitive* method is used. Negative training instances are generated by pairing an NP with each of its preceding non-coreferent NPs. During clustering, NPs that are classified as coreferent are merged. Positive classifications are given a higher precedence than negative classifications when conflicts arise. Finally, McCarthy and Lehnert report that the learned coreference classifier outperforms a rule-based coreference classifier, a result that is consistent with that of Aone and Bennett.

Soon et al.'s knowledge-lean coreference system. Soon et al. (2001) adopt a knowledge-lean approach to general-purpose coreference resolution in which the decision tree learning algorithm has access to only 12 surface-level features. By contrast, the 12 features are all designed to be domain-independent, including one lexical feature (string matching), eight grammatical features (gender and number agreement, apposition, and NP types), two semantic features (semantic class compatibility and aliasing), and one positional feature (number of sentences between the two NPs). Unlike McCarthy and Lehnert, the *non-transitive* method is used to generate positive training instances from coreference chains. To reduce the ratio of negative instances to positive instances (and hence the skewness of the class distributions), only a subset of the available negative instances is used for training: a negative instance is created by pairing an anaphoric NP, NP_j, with each NP appearing between NP_j and its closest preceding antecedent in the associated text. A left-to-right single-link clustering algorithm that selects the closest preceding coreferent NP as the antecedent for each anaphoric noun phrase is employed to create coreference equivalence classes. Results indicate that STR-MATCH (i.e. whether the surface strings of the two NPs are the same after determiners

[6] When an anaphor has more than one antecedent according to the coreference classifier, the antecedent that has the highest confidence value is selected during clustering.

are discarded), ALIAS (i.e. whether two named entities are aliases)[7], and APPOSITIVE (i.e. whether the two NPs form an appositive construction) are strong indicators of coreference. The system is applied to two standard coreference data sets (MUC-6, 1995, MUC-7, 1998), achieving performance comparable to the best-performing knowledge-based coreference engines.

Kehler's work on probablistic coreference. Rather than performing coreference resolution before generating templates in a standard IE architecture (Cardie, 1997), Kehler's coreference system (Kehler, 1997b) operates on the output of an IE system in which one template is produced for each entity mentioned in the document. The goal of Kehler's system is then to generate a probability distribution over all possible partitions of the output templates such that the templates in each equivalence class are coreferent. The coreference system derives a maximum entropy model for determining the probability that two templates co-refer based on three types of features. The first type of feature tests whether two templates possess a subsumption relationship by checking whether the relationship is satisfied for each slot in the template. The second type of feature is based on the definiteness of the entity associated with one of the templates. The third type of feature measures the distance between the entities associated with the templates. Training instances are generated from all pairs of templates. The system then uses *Dempster's Rule of Combination* (Dempster, 1968) to combine the pairwise coreference probabilities generated from the learned model to score a partition. The resulting scores (with one score per partition) are normalized to form a probability distribution over the set of possible partitions. Because the number of possible partitions grows exponentially with the number of templates[8], it is unclear whether Kehler's approach is scalable to a large number of templates.

Mueller et al.'s co-training method for reference resolution. One potential disadvantage of the supervised learning approaches described in this subsection is that they require a large amount of manually annotated data. While corpora annotated with anaphora or coreference information exist for English, they are not readily available for the majority of the languages. This has prompted the development of *weakly supervised* approaches for the task. For example, Mueller et al. (2002) use co-training (Blum and Mitchell, 1998) to combine labeled and unlabeled data to train a classifier for resolving anaphoric references in German. Given a set of 16 features based on Soon et al. (2001) and Cardie and Wagstaff (1999), Mueller et al. present an algorithm for partitioning the feature set into two views, each of which is used to train a decision tree classifier for resolving anaphoric references. The initial training and test sets are formed by pairing each NP with each of its preceding NPs. Two steps of example selection are performed on both data sets.[9] First, a number of heuristics are used to remove negative instances in order to reduce the skewness of class

[7]Here, different types of aliasing are defined for different named-entity types: Two dates are aliases if their normalized form is identical. Two persons are aliases if the last word of the two NPs involved are identical. Two organizations are aliases if one is an acronym of the other.

[8]According to Anderberg (1973), the number of ways to partition n items is

$$\sum_{k=1}^{n} S_n^{(k)}$$

where

$$S_n^{(k)} = \frac{1}{k!}\sum_{i=0}^{k}(-1)^{k-i}\binom{k}{i} i^n.$$

[9]This is primarily to guarantee that the instances used for training and testing are drawn from the same distribution.

distributions. Second, the Soon et al. method of example selection is applied, thus ensuring that exactly one positive instance is formed from each anaphoric expression. Each data set is further split into three disjoint subsets according to the feature that determines the anaphor type: personal and possessive pronouns, proper names, and definite NPs. Results are reported in terms of F-measure for each type of anaphora. In contrast to previous results on co-training, their results show that for all types of anaphora except definite NPs, the classifiers that are trained on both labeled and unlabeled data do not yield considerable improvement over those that use only labeled data. They then conclude that co-training might only be useful for improving resolution performance on certain types of anaphora. However, they do not provide any analysis of why the task does not benefit from unlabeled data. In particular, the degree to which the underlying assumptions of co-training (i.e. view independence and compatibility) are satisfied by the data sets is not checked at all. Recent work by Muslea et al. (2002) have demonstrated that co-training is not robust to the violations of the two assumptions. Consequently, further investigations are needed to determine whether the co-training results obtained by Mueller et al. can be accounted for along these lines.

4.3.3 Statistical Approaches

As a subclass of corpus-based approaches, statistical approaches to anaphora resolution attempt to construct a model that determines the probability that an NP is an antecedent of a given anaphor. The general approach here is to decompose the equation for the model, which is initially conditioned on a number of parameters, into pieces that correspond to factors affecting anaphora resolution by making use of Bayes rule and certain independence assumptions.

Ge et al.'s statistical approach. Ge et al. (1998) present a probability model for resolving third-person pronouns. Four training features for the model are used, including positional information (i.e. the distance between the pronoun and the candidate antecedent), grammatical information (i.e. gender and animacity of the candidate antecedent), semantic information (i.e. selectional preferences based on the governing constituent of the pronoun), and a crude measure of salience (i.e. mention count of the candidate antecedent). Ge et al. show how the equation for the model can be decomposed into these four factors for which statistics can be collected directly from the training corpus. For a given anaphoric pronoun, the candidate antecedent that is assigned the highest probability by the model is selected as the antecedent of the pronoun. Their results demonstrate that knowledge of the candidate antecedent's gender and animacity and its salience are essential to anaphora resolution.

So far we have provided an overview of previous work on anaphora and coreference resolution and characterized each system as either knowledge-based or corpus-based. Instead of performing a similar characterization on a system basis along the remaining dimensions, we will point out some of the key issues involved in each of these dimensions and focus on the strengths and weaknesses of the related approaches. In section 4.4, we will compare knowledge-rich approaches with knowledge-lean approaches. In section 4.5, we will look at the implications of semi-automatic pre-processing and fully automatic pre-processing. Section 4.6 discusses the issues with respect to small-scale evaluation versus large-scale evaluation of anaphora and coreference resolution systems.

4.4 Knowledge-Rich Approaches versus Knowledge-Lean Approaches

As we mentioned above, anaphora and coreference resolution systems can in principle be ordered according to the *types* of knowledge source required to perform accurate resolution: knowledge-rich or knowledge-lean. Early anaphora resolution systems such as Grosz (1977), Sidner (1979), and Carbonell and Brown (1988) are knowledge-rich systems that rely on domain information as well as semantic and pragmatic knowledge that is often hard to represent and process and needs to be encoded for each new domain. Knowledge-lean approaches, which only makes use of morphological and shallow syntactic information, are developed precisely to meet the need for efficient solutions to anaphora resolution. In particular, knowledge-lean anaphora resolution systems such as Lappin and Leass (1994), Kennedy and Boguraev (1996), Baldwin (1997) and Mitkov (1998) have all demonstrated that high-performance anaphora resolution is possible without semantic and world knowledge. Although Lappin and Leass's algorithm assumes the output of a full parser, Kennedy and Boguraev show that the system only suffers from a small drop in accuracy when the parser output is approximated using a set of heuristics.

Given the successful shift from knowledge-rich approaches to knowledge-lean approaches for the task of anaphora resolution in recent years, two immediate questions arise:

- How much extra benefit can be obtained if the system has access to semantic information? Equivalently, how much performance loss is incurred by relying only on morphological and shallow syntactic information?
- Does the same result generalize to coreference resolution? In other words, is high-performance knowledge-poor coreference resolution possible?

The work by Ge et al. (1998) sheds some light on the first question. Specifically, one of the factors that their statistical pronoun resolution system depends on is selectional restrictions, which is a form of semantic knowledge. They evaluate their system by incorporating the knowledge sources incrementally into the system. Their results indicate that the system that has access to semantic knowledge only performs marginally better than the one without semantic knowledge. However, they also remark that selectional restrictions "are not clearcut in most cases" and that "some verbs are too general to restrict the selection of any NP".

On the other hand, the answer to the second question seems to be different under different circumstances. For domain-specific coreference resolution, both Aone and Bennett (1995) and McCarthy and Lehnert (1995) report that the decision tree classifiers take advantage of domain-specific semantic information. In contrast, the results of Soon et al. (2001) seem to imply that a general-purpose coreference resolution can achieve good performance without semantic information. Specifically, their decision tree classifier relies almost entirely on string matching, apposition, and aliasing for recognizing coreference relationships, despite the fact that it has access to information regarding semantic class compatibility. Although aliasing is generally viewed as a semantic knowledge source, the corresponding feature value is actually computed via a set of simple lexical matching routines without resorting to any semantic knowledge base. Nevertheless, the fact that the semantic feature is not selected by Soon's decision tree is by no means an indication that semantic information is irrelevant to coreference resolution, since the error rate resulting from the automatic computation of the feature has not been measured.

Despite the popularity of knowledge-lean approaches, many of the MUC coreference systems make use of semantic information provided by WordNet (Fellbaum, 1998). In general, the availability of general-purpose lexical resources such as WordNet and advances in research in named-entity classification have made it convenient for anaphora and coreference resolution systems to incorporate semantic information. It should be noted, however, that systems using semantic information from WordNet need to somehow deal with the word sense disambiguation problem, which is widely known to be a hard problem in NLP. Consequently, the issue regarding the accuracy of semantic feature computation remains.

4.5 Semi-Automatic Pre-Processing versus Fully Automatic Pre-Processing

Mitkov (2001) points out that research in anaphora resolution has "suffered from a bizarre anomaly in that until recently hardly any fully automatic operational systems had been reported". For example, Lappin and Leass (1994) post-edit the outputs of the part-of-speech tagger and the full parser on which their anaphora resolution algorithm relies. Similarly, Mitkov (1998) manually fixes the outputs of the part-of-speech tagger and the base noun phrase finder. In addition, the knowledge assumed by the resolution algorithms described in Mitkov (1997) such as gender is obtained directly from the data sets that are annotated with the corresponding information. Some exceptions exist, however. The BFP anaphora resolution algorithm (Brennan et al., 1987) is fully automatic, for example; the gender and number information required by their centering algorithm is provided by other modules in their HPSG natural language system.

The MUC-6 and MUC-7 conferences (MUC-6, 1995, MUC-7, 1998) have promoted the development of coreference resolution systems that operate in fully automatic mode. In addition, virtually all coreference systems developed since MUC-6 (e.g. Aone and Bennett (1995), McCarthy and Lehnert (1995)[10], Soon et al. (2001)) as well as the CogNIAC pronoun resolution system (Baldwin, 1997), which was part of the CAMP system participating in the MUC coreference task (Baldwin et al., 1998), are also fully automatic. Furthermore, the emergence of statistical approaches has obviated the need for certain kinds of preprocessing. For example, Ge et al. (1998) describe a method for automatically computing gender information from an annotated corpus.

The reason for the shift from semi-automatic pre-processing to fully automatic-preprocessing is that the former is practically unrealistic. Specifically, it is impractical to ask a human to stand by and fix all errors made by pre-processing routines. Even if the pre-processing taggers are all highly reliable and only a small number of errors needs to be fixed manually, the human still has to go through all the outputs of the taggers in order to locate the errors. This proves to be a time-consuming task, especially if the system relies on a large number of knowledge sources or needs to process a large number of documents.

On the other hand, feeding an anaphora or coreference algorithm with perfectly computed knowledge has the advantage that the performance of the resolution algorithm is not affected by errors propagated from upstream pre-processing components. Consequently, we can easily measure the errors made by the resolution algorithm versus the errors made by the pre-processing routines.

Recent work has shown that the performance of resolution algorithms is severely limited by the accuracy with which the knowledge required by the algorithms can be computed, thus providing suggestive evidence that the task has been made much easier in the presence of semi-automatic pre-processing. In particular, a recent re-implementation of Mitkov's robust pronoun resolution algorithm (Mitkov, 1998) and evaluation on new data sets by Mitkov et al. (2002) shows that the performance of the new, fully automatic algorithm is not as high as the initial results obtained using the semi-automatic algorithm. As a result, it makes sense to take into account the *robustness* of a resolution algorithm to potentially erroneous input when measuring its performance. As a side note, there are two classes of algorithms that are in principle more robust than the others:

- Knowledge-lean algorithms: As noted in section 2, an anaphora or coreference resolution algorithm can potentially rely on a large number of linguistic knowledge sources. With respect to the accuracy of knowledge sources, lower-level knowledge such as lexical and syntactic knowledge can be computed with comparatively better accuracies than higher-level

[10]The feature values provided to this coreference system are all computed manually. The authors claim that this makes it possible to evaluate the errors made by the coreference resolution *algorithm* and that errors made by pre-processing routines should be evaluated separately.

knowledge such as semantic and pragmatic knowledge. In fact, except possibly for lexical string matching which can be performed without any error, none of the knowledge sources can be computed with perfect accuracy. Consequently, knowledge-lean resolution algorithms that usually depend on just the outputs of a part-of-speech tagger and a base noun phrase finder (e.g. Baldwin (1997), Mitkov (1998)) tend to be more robust than their knowledge-rich counterparts, especially since the corresponding tasks can be accomplished with fairly high accuracies.

- Corpus-based algorithms: Corpus-based algorithms are trained on real-world texts and hence are by nature more robust in the face of incomplete and noisy input than their knowledge-based counterparts. For example, we mentioned in section 2 that although there are exceptions to traditional linguistic constraints such as number agreement, many knowledge-based systems such as Lappin and Leass (1994) and Kennedy and Boguraev (1996) simply encode them as broadly and unconditionally hard constraints and thus effectively ignore the exceptions. In contrast, corpus-based algorithms learn soft constraints empirically from labeled data and can therefore take into account these exceptions.

4.6 Small-Scale Evaluation versus Large-Scale Evaluation

For most early anaphora resolution algorithms, evaluation is done by hand on a carefully chosen small set of documents, due in part to the fact that these algorithms were not implemented as computer programs at the time of evaluation. For example, Hobbs evaluates his algorithm (Hobbs, 1978) on 300 pronouns extracted from three different sources. The comparative evaluation by Walker (1989) on the Hobbs algorithm and the BFP algorithm is performed on 281 third-person pronouns in articles from three different genres.

One advantage of manual evaluation on small data sets over automatic evaluation is that it offers a better picture of what mistakes are made by an algorithm and how the algorithm can be improved. On the other hand, evaluation on small data sets means that the resulting performance of the algorithm may not accurately reflect its actual performance, at least from a statistical perspective. The recent explosion of the number of online texts has made large-scale, automatic evaluation possible.

5 Improving Machine Learning Approaches

In the previous section, we have seen how existing anaphora and coreference resolution systems can be characterized along four different dimensions: knowledge-based approaches versus corpus-based approaches, knowledge-rich approaches versus knowledge-lean approaches, semi-automatic pre-processing versus fully automatic pre-processing, and small-scale evaluation versus large-scale evaluation. Although state-of-the-art coreference resolution systems are mostly knowledge-based systems (e.g. all but one systems participating in the MUC-6 coreference task were knowledge-based, and all participating in MUC-7 were knowledge-based), machine learning techniques have been applied reasonably successfully to the problem. For example, Soon et al. (2001) apply an NP coreference system based on decision tree induction to the MUC-6 and MUC-7 coreference data sets, achieving performance that lags the best results (produced by knowledge-based coreference engines) only by 2.3% and 1.7% in F-measure.

As noted in section 1.2, we propose to investigate whether existing machine learning approaches to coreference resolution can be improved. Our decision to focus on machine learning approaches to the problem stems from two observations:

Learning-based coreference systems have the potential to outperform their knowledge-based counterparts. In principle, learning-based systems are more robust than knowledge-based systems in the face of noisy input. As we have seen before, the input to a coreference system is noisy, since most of the knowledge sources that the system potentially relies on cannot be computed with high accuracies. Even knowledge-lean systems that rely on a small number of knowledge sources that can be computed with fairly high accuracies are not as robust as they claim. For example, Mitkov et al. (2002) show that replacing manually computed knowledge sources with their automatically computed counterparts precipitates a statistically significant drop in the performance of their pronoun resolution system. In addition, learning-based systems are better at handling exceptions, as noted above, and can selectively incorporate those knowledge sources that are found to be relevant and useful. Furthermore, there is empirical evidence that learning-based systems can perform better than knowledge-based systems. Both Aone and Bennett (1995) and McCarthy and Lehnert (1995) report that their learned coreference classifier outperforms their rule-based classifier.

Existing learning-based coreference systems are far from perfect. For example, Soon et al.'s learning-based coreference system achieves an F-measure of 62.6% and 60.4% on the MUC-6 and MUC-7 coreference data sets respectively, indicating that there is still substantial room for improvement.

Since we approach coreference resolution as a learning-based NLP problem, there are two orthogonal directions in which extensions can naturally be made:

Linguistically motivated extensions. This type of extensions takes into account what linguistics theories can offer regarding discourse anaphora and coreference and hence is problem-specific. Specifically, we consider the following dimensions:

- Feature set enhancement. In section 5.2, we augment the feature set commonly employed by learning-based coreference systems in terms of both *number* and *sophistication* using our linguistic intuitions regarding coreference. This modification in principle allows the learning algorithm to acquire a better model.
- Anaphoricity determination. This is basically step 3 of the GENERIC-RESOLVE algorithm. As noted in section 3, anaphoricity determination can reduce the amount of processing that needs to be performed by the resolution algorithm and can potentially increase the precision of the system. Nevertheless, none of the existing learning-based coreference systems has attempted to make use of anaphoricity information explicitly. Section 5.3 presents a new approach to anaphoricity determination and evaluates the contribution of anaphoricity information to coreference resolution.

Extra-linguistic extensions. This type of extensions essentially aims at improving existing machine learning framework for coreference resolution. Some of them are applicable to learning problems in general, whereas others are specific to coreference resolution. More precisely, we consider the following dimensions:

- Precision-enhancing modifications. As a first extension, we propose several simple but natural modifications to the standard machine learning framework that aim to improve the precision of the coreference system. Since we view coreference as a problem of selecting an antecedent for each potentially anaphoric NP from a confusion set, it seems most natural to pick the element in the set with which we have the highest confidence. We discuss this confidence-based approach in more detail in section 5.4.

- Feature selection. In many machine learning problems, the feature set is usually composed of both relevant and irrelevant features. Feature selection is concerned with selecting a subset of the features in the feature set that are relevant to the problem. The ability to select a "good" set of features often results in better performance. We will examine this issue in section 5.5.

- Training instance selection. Training instance selection is concerned with selecting a smaller subset of instances from the set of available training instances. Section 5.6 explains the need for instance selection based on two intrinsic properties of coreference.

- Rule selection. Many learning algorithms (such as decision trees and rule learning algorithms) have the capability of "pruning" the learned model in hopes of preventing overfitting and improving generalization performance. Rule selection is concerned with selecting a subset of rules from a learned model that can be transformed into an equivalent ruleset. Section 5.7 motivates why an explicit rule selection step can potentially offer performance improvements.

For the rest of the section, we describe the work we have done towards the goal so far and outline what we plan to do for the rest of the dissertation. We begin in section 5.1 by introducing our Baseline coreference system, which is a re-implementation of the best existing learning-based coreference system (Soon et al., 2001). Then we describe in more detail the dimensions of improvement to this basic machine learning framework that we mentioned above. Specifically, we give a high-level overview of the additional features we use for coreference in section 5.2. In section 5.3, we present a new approach to anaphoricity determination. We explain three simple precision-enhancing modifications to the standard machine learning framework in section 5.4. Sections 5.5, 5.6, and 5.7 discuss feature selection, training instance selection, and rule selection respectively. We conclude this section with our proposed work for the dissertation.

For each dimension of improvement, we will describe the progress we have made so far and present some initial results that are based on our recent publications. Since these papers are written with slightly different goals, the baseline systems to which the corresponding results are compared are not identical across all dimensions, although these baseline systems are either the same as or variants of the one that will be described in section 5.1. Consequently, one of our proposed work is to incorporate the changes entailed by these dimensions incrementally into the Duplicated Soon Baseline system.

5.1 The Baseline Coreference System

Our baseline coreference system attempts to duplicate both the approach and the knowledge sources employed in Soon et al. (2001), which was described briefly in section 4.3.2. More specifically, the baseline system employs the standard combination of classification and clustering described above.

Creating an instance. An instance in our baseline system is a description of two NPs in a document. More formally, let NP_k be the kth NP in a document. An instance formed from NP_i and NP_j is denoted by $i_{(NP_i,NP_j)}$. A *valid* instance is an instance $i_{(NP_i,NP_j)}$ such that NP_i precedes NP_j.[11]

[11] By definition, exactly $\binom{n}{2}$ valid instances can implies that at most $\binom{n}{2}$ distinct instances can be created from n NPs in a given document.

Feature Type	Feature	Description
Lexical	SOON_STR	C if, after discarding determiners, the string denoting NP_i matches that of NP_j; else I.
Grammatical	PRONOUN_1	Y if NP_i is a pronoun; else N.
	PRONOUN_2	Y if NP_j is a pronoun; else N.
	DEFINITE_2	Y if NP_j starts with the word "the;" else N.
	DEMONSTRATIVE_2	Y if NP_j starts with a demonstrative such as "this," "that," "these," or "those;" else N.
	NUMBER	C if the NP pair agree in number; I if they disagree; NA if number information for one or both NPs cannot be determined.
	GENDER	C if the NP pair agree in gender; I if they disagree; NA if gender information for one or both NPs cannot be determined.
	BOTH_PROPER_NOUNS	C if both NPs are proper names; NA if exactly one NP is a proper name; else I.
	APPOSITIVE	C if the NPs are in an appositive relationship; else I.
Semantic	WNCLASS	C if the NPs have the same WordNet semantic class; I if they don't; NA if the semantic class information for one or both NPs cannot be determined.
	ALIAS	C if one NP is an alias of the other; else I.
Positional	SENTNUM	Distance between the NPs in terms of the number of sentences.

Table 1: Feature Set for the Duplicated Soon Baseline system. The feature set contains relational and non-relational features. Non-relational features test some property P of one of the NPs under consideration and take on a value of **YES** or **NO** depending on whether P holds. Relational features test whether some property P holds for the NP pair under consideration and indicate whether the NPs are **COMPATIBLE** or **INCOMPATIBLE** w.r.t. P; a value of **NOT APPLICABLE** is used when property P does not apply.

Following previous work (Aone and Bennett (1995), Soon et al. (2001)), we assume throughout that only valid instances will be generated and used for training and testing. Each training instance consists of the 12 Soon et al. features, which are described in Table 1. Linguistically, the features can be divided into four groups: lexical, grammatical, semantic, and positional.[12] The classification associated with a training instance is one of COREFERENT or NOT COREFERENT depending on whether the NPs co-refer in the associated training text.

Building an NP coreference classifier. We use the C4.5 decision tree induction system (Quinlan, 1993) to train a classifier that, given a test instance $i_{(NP_i, NP_j)}$, decides whether or not NP_i and NP_j are coreferent. We follow the procedure employed in Soon et al. to create the training data: we rely on coreference chains from the MUC answer keys to create (1) a *positive instance* for each anaphoric noun phrase, NP_j, and its closest preceding antecedent, NP_i; and (2) a *negative instance* for NP_j paired with each of the intervening NPs, NP_{i+1}, NP_{i+2},. . ., NP_{j-1}. This method of negative instance selection is further described in Soon et al. (2001); it is designed to operate in conjunction with their method for creating coreference chains.

Applying the classifier to create coreference chains. After training, the decision tree is used by a clustering algorithm to impose a partitioning on all NPs in the test texts, creating one cluster for each set of coreferent NPs. As in Soon et al., texts are processed from left to right. Each NP encountered, NP_j, is compared in turn to each preceding NP, NP_i, from right to left. For each pair, a test instance is created as during training and is presented to the coreference

[12]In all of the work presented here, NPs are identified, and features values computed entirely automatically.

classifier, which returns a number between 0 and 1 that indicates the likelihood that the two NPs are coreferent.[13] NP pairs with class values above 0.5 are considered COREFERENT; otherwise the pair is considered NOT COREFERENT. The process terminates as soon as an antecedent is found for NP_j or the beginning of the text is reached.

In the rest of the section, we will investigate whether we can make improvements to the Baseline system. Unless otherwise stated, all system variations are evaluated using the model-theoretic MUC scoring program (Vilain et al., 1995) and performance reported in terms of recall, precision, and F-measure on the standard MUC-6 and MUC-7 coreference corpora, training the coreference classifier on the 30 "dry run" texts, and applying the coreference resolution algorithm on the 20–30 "formal evaluation" texts.[14] Except for sections 5.6 and 5.7 in which we use RIPPER (Cohen, 1995) as our underlying learning algorithm, all of the results shown below are obtained by training a coreference classifier using C4.5.

5.2 Feature Set Enhancement

This subsection focuses on linguistic-related issues for coreference resolution. In particular, we are interested in knowing whether incorporating additional linguistic knowledge can improve the performance of a corpus-based coreference resolution algorithm. As we mentioned before, much work has been done on investigating the knowledge sources that are important for coreference resolution. However, the investigation has proceeded to a large extent in the context of knowledge-based approaches to coreference resolution, which implies that linguistic constraints are generally treated as broadly and unconditionally applicable hard constraints. In addition, it is difficult to evaluate the relative contribution of the available knowledge sources in knowledge-based systems. On the other hand, most learning-based systems have relied on a fairly small number of features (usually between 5 and 12), with the exception of Aone and Bennett's coreference system (Aone and Bennett, 1995), which makes use of 66 features. Unfortunately, Aone and Bennett do not report the features employed by their system. Consequently, the question of whether increasing the *number* and *sophistication* of the features can improve the performance of a learning-based coreference system remains.

Work to date. We have expanded the Soon et al. feature set from 12 features to an arguably deeper set of 53 (Ng and Cardie, 2002c). We propose additional lexical, semantic, and knowledge-based features; most notably, however, we propose 26 additional grammatical features that include a wide variety of linguistic constraints and preferences. The features were not derived empirically from the corpus, but were based on common-sense knowledge and linguistic intuitions regarding coreference.

Results. Contrary to our expectation that system performance will increase with the incorporation of additional knowledge, our experimental results show that F-measure drops precipitously for both the MUC-6 and MUC-7 data sets when compared to the Baseline coreference system. In

[13]We convert the binary class value using the smoothed ratio $\frac{p+1}{t+2}$, where p is the number of positive instances and t is the total number of instances contained in the corresponding leaf node.

[14]The MUC scoring program reports the performance of a coreference resolution system in terms of recall, precision, and F-measure. Recall and precision are measures of task coverage and accuracy respectively, and F-measure is the balanced harmonic mean of these measures.

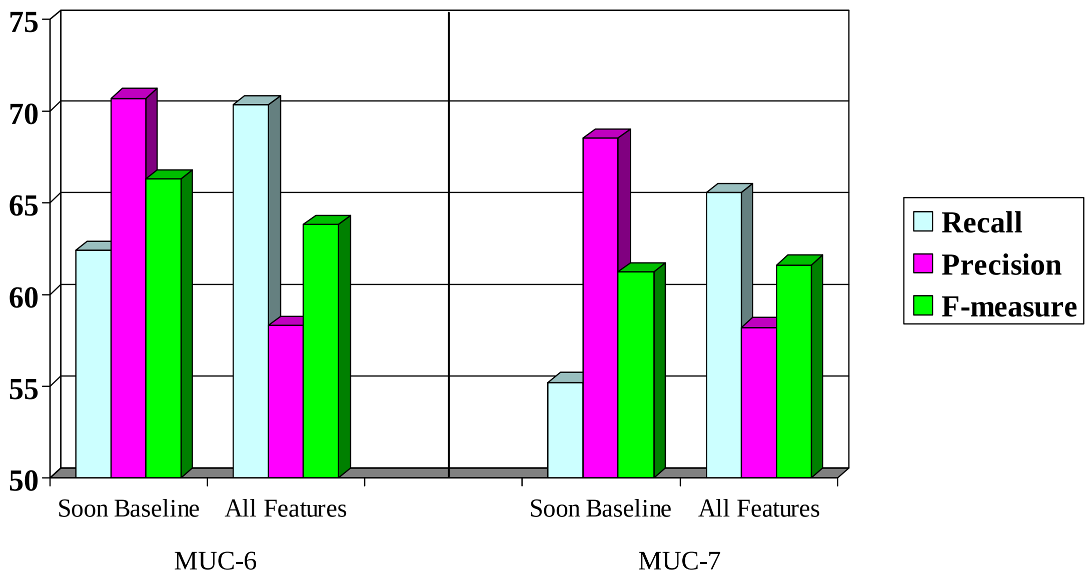

Figure 3: Effects of Incorporating Additional Knowledge into the Coreference Resolution System

particular, we see statistically significant increases in recall, but much larger decreases in precision.[15] (See Figure 3.) A closer examination of the results indicate very poor precision on common nouns in comparison to that of pronouns and proper nouns. In particular, the classifiers acquire a number of low-precision rules for common noun resolution, presumably because the current feature set consists of irrelevant features or is insufficient for the classification. One potential solution to this problem is feature selection, which we will address in section 5.5.

5.3 Anaphoricity Determination

As we have seen in section 3, anaphoricity determination is the task of determining whether a given discourse entity is anaphoric or not. Recall that in the Baseline system, a single-link clustering algorithm performs a right-to-left search for an antecedent for each discourse entity. Consequently, the availability of anaphoricity information can potentially help coreference resolution in the following ways:

- Save processing time: This is because the clustering algorithm will not be spending time on searching for an antecedent for non-anaphoric noun phrases. This savings could be huge. For example, about 70% of the noun phrases in the MUC data sets are not anaphoric.
- Enhance precision: This is because any antecedent that is being found for a non-anaphoric noun phrase is certainly wrong and will therefore damage the precision of the coreference resolution system.

In fact, a number of researchers have tackled the problem of anaphoricity determination (e.g. Paice and Husk (1987), Lappin and Leass (1994), Kennedy and Boguraev (1996), Denber (1998), Bean and Riloff (1999), Vieira and Poesio (2000), Evans (2001)). Nevertheless, our supervised

[15]Chi-square statistical significance tests are applied to changes in recall and precision throughout the rest of the paper. Unless otherwise noted, reported differences are at the 0.05 level or higher. The chi-square test is not applicable to F-measure.

learning approach to anaphoricity determination (Ng and Cardie, 2002b) has a number of advantages over existing approaches:

- *The approach is supervised, so no hand-coded heuristics are required.* In contrast, the anaphoricity determination algorithms proposed by Paice and Husk (1987), Lappin and Leass (1994), Kennedy and Boguraev (1996), Denber (1998), and Vieira and Poesio (2000) are heuristic-based.

- *All of the features used (and hence the resulting classifier) are domain-independent.* Bean and Riloff (1999) present an unsupervised approach for constructing a list of non-anaphoric entities from an unannotated corpus. Although their approach is domain-independent, the strength of the approach lies in the use of lexical information for acquiring a list of non-anaphoric entities consisting mainly of domain-specific terms, which is useful for classifying documents from that domain.

- *The approach allows additional insight into the factors important to anaphoricity determination from an information-theoretic perspective.* In contrast, it is often not easy to evaluate the relative contribution of each knowledge source available to the problem with heuristic-based approaches (e.g. Paice and Husk (1987), Lappin and Leass (1994), Kennedy and Boguraev (1996), Denber (1998), Vieira and Poesio (2000)).

- *The approach performs anaphoricity determination on all types of NPs.* Some anaphoricity determination algorithms address only pleonastic pronouns (e.g. Paice and Husk (1987), Lappin and Leass (1994), Kennedy and Boguraev (1996), Denber (1998)), while others focus on identifying anaphoric and non-anaphoric uses of *it* (e.g. Evans (2001)) and definite descriptions (e.g. Bean and Riloff (1999), Vieira and Poesio (2000)). In general, anaphoricity determination for non-definites is largely ignored by previous approaches.[16] We found, however, that a non-negligible number (7% and 13%) of the non-definite NPs in the MUC-6 and MUC-7 data sets (MUC-6, 1995, MUC-7, 1998), respectively, are anaphoric.

Work to date. As we mentioned above, we have adopted a supervised learning approach to anaphoricity determination (Ng and Cardie, 2002b). The anaphoricity determination component can be trained using standard supervised machine learning methods as described below.

Building a classifier for anaphoricity determination. We use the C4.5 decision tree induction system to train a classifier that, given a description of an NP in a document, decides whether or not the NP is anaphoric. Each training instance represents a single NP and consists of 37 features that are potentially useful for distinguishing anaphoric and non-anaphoric NPs. the classification associated with a training instance — one of ANAPHORIC or NOT ANAPHORIC is obtained from coreference chains in the training texts. Specifically, a *positive instance* is created for each NP that is involved in a coreference chain but is not the head of a chain. A *negative instance* is created for each of the remaining NPs.

Applying the classifier. To determine the anaphoricity of an NP in a test document, we create an instance for it as during training and present it to the decision tree anaphoricity classifier, which returns a value of ANAPHORIC or NOT ANAPHORIC.

[16]We consider an NP to be a non-definite if it is neither a pronoun nor a proper name and does not start with the article "the" or a demonstrative such as "this", "that", "these" or "those".

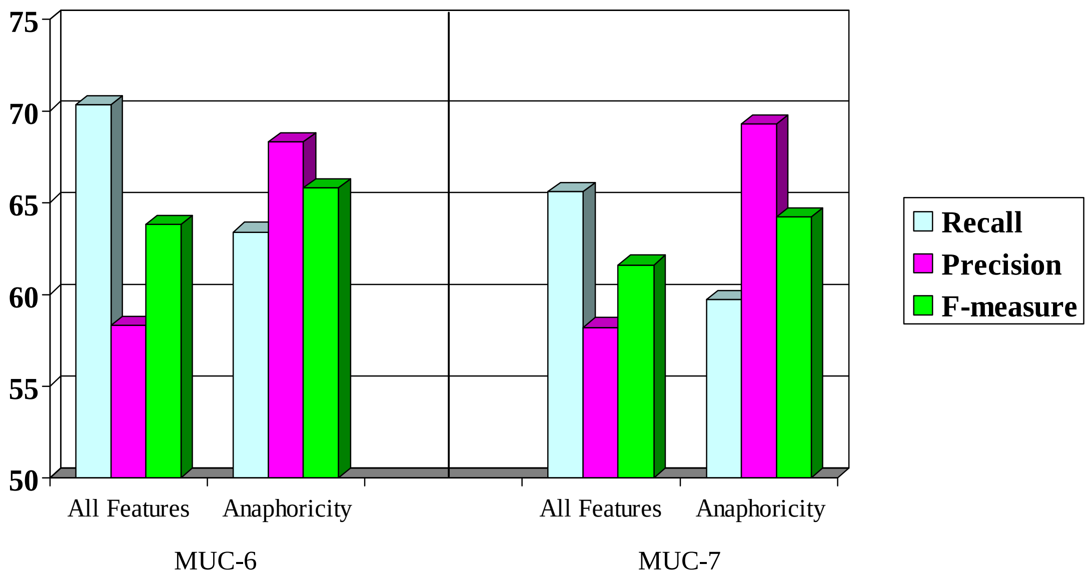

Figure 4: Effects of Anaphoricity Determination on Coreference Resolution

Results. Our results using anaphoricity determination for coreference resolution are mixed. In particular, we find that incorporating anaphoricity information causes performance to drop significantly when compared to the coreference resolution system using the full feature set consisting of all 53 features. (Note that these results are not shown.) Unlike Mitkov et al. (2002) who attribute the drop to the inability of the scoring function to capture the performance of the anaphoricity classifier, we instead explain the phenomenon in terms of the (in)accuracy of the anaphoricity classifier and show that the drop is particularly dramatic for common nouns in contrast to pronouns and proper nouns. Augmenting the anaphoricity classifier with two heuristics that encode prior knowledge regarding common noun resolution corrects this problem, and produces a coreference resolution system that can benefit from the availability of anaphoricity information. Figure 4 compares the performance of the two coreference systems using the full feature set, where one does not have access to anaphoricity determination and the other is used in conjunction with an anaphoricity classifier augmented with the two heuristics mentioned above.

5.4 Precision-enhancing Modifications

Existing work on anaphora and coreference resolution has generally focused on the development of knowledge sources that are potentially useful for the task. After examining some of these linguistic issues, we now turn to modifications to the standard machine learning framework for coreference resolution in this and the next three subsections. Our work on non-linguistic modifications here is partly motivated by Mitkov's observation that extra-linguistic strategies also play an important role in the performance of an anaphora resolution algorithm (Mitkov, 1997).

Work to date. We experimented with three simple but natural precision-enhancing modifications to the Soon et al. machine learning framework, with the goal of improving precision in the resulting coreference resolution system (Ng and Cardie, 2002c). First, we modify the clustering algorithm in order to make the *most confident* decisions, a strategy that is well justified when reasoning under uncertainty. The second extension involves training instance selection, a modification that is a natural consequence of the first one. The last modification involves replacing a feature with several primitive features in hope to give the learning algorithm additional flexibility in creating coreference classifiers. The three precision-enhancing modifications are described in more detail below:

- Best-first clustering. Rather than a right-to-left search from each anaphoric NP for the first coreferent NP, we hypothesized that a right-to-left search for a *highly likely antecedent* might offer more precise, if not generally better coreference chains. As a result, we modify the coreference clustering algorithm to select as the antecedent of NP_j the NP with the highest coreference likelihood value from among preceding NPs with coreference class values above 0.5.

- Training set creation. For the proposed best-first clustering to be successful, however, a different method for training instance selection would be needed: rather than generate a positive training example for each anaphoric NP and its **closest** antecedent, we instead generate a positive training example for its **most confident** antecedent. More specifically, for a non-pronominal NP, we assume that the most confident antecedent is the closest **non-pronominal** preceding antecedent. For pronouns, we assume that the most confident antecedent is simply its closest preceding antecedent. Negative examples are generated as in the Baseline system.

- String match feature. Soon's string match feature (SOON_STR) tests whether the two NPs under consideration are the same string after removing determiners from each. We hypothesized, however, that splitting this feature into several primitive features, depending on the type of NP, might give the learning algorithm additional flexibility in creating coreference rules. Exact string match is likely to be a better coreference predictor for proper names than it is for pronouns, for example. Specifically, we replace the SOON_STR feature with three features — PRO_STR, PN_STR, and WORDS_STR — which restrict the application of string matching to pronouns, proper names, and non-pronominal NPs, respectively.

Results. Experimental results indicate that when the three modifications are used in combination, they consistently provide statistically significant gains in precision over the Duplicated Soon Baseline system without any loss in recall. As a result, we observe reasonable increases in F-measure. (See Figure 5). Similar, but weaker, effects occur when applying each of the learning framework modifications to the Baseline system in isolation (not shown in the figure).

5.5 Feature Selection

In section 5.2, we have seen that system performance deteriorates with the incorporation of additional knowledge, as a result of the induction of low-precision rules. This can partly be attributed to the presence of irrelevant features in the feature set, thus making feature selection an issue for learning-based coreference resolution systems. Although decision tree learners are known to have an implicit feature selection capability, previous research has shown that feature selection can improve the performance of decision tree classifiers substantially on certain data sets (e.g. Almuallim and Dietterich (1991), John et al. (1994)). Feature selection for coreference resolution has not been addressed in the literature, presumably because most existing learning-based coreference systems rely on a fairly small number of features and consequently feature irrelevance may pose less of a problem for those systems.

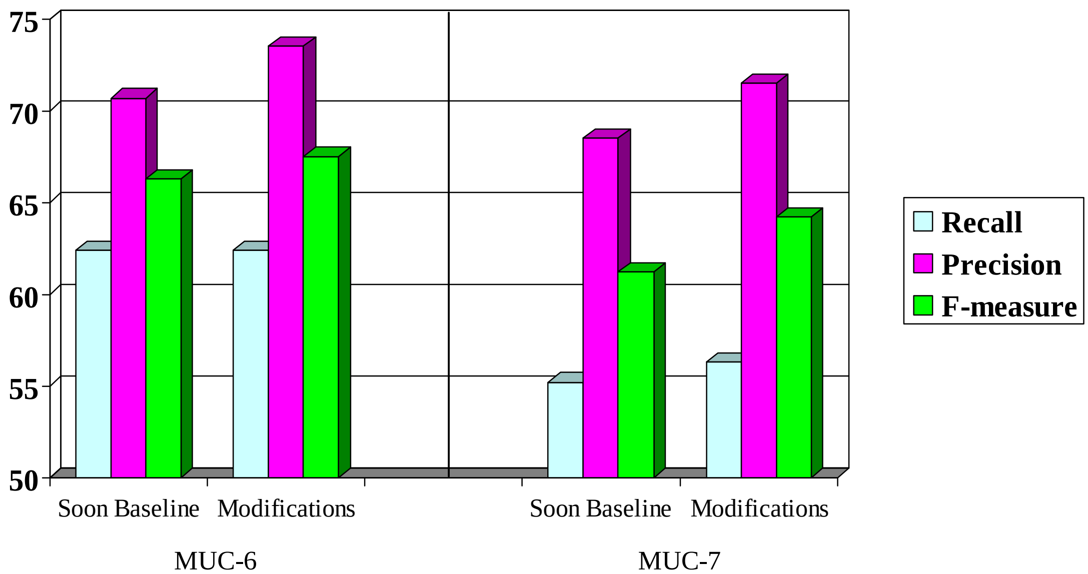

Figure 5: Effects of Precision-Enhancing Modifications

Work to date. We have evaluated a version of the coreference system that employs manual feature selection (Ng and Cardie, 2002c): for each data set, we discard features used primarily to induce the low-precision rules for common noun resolution and re-train the coreference classifier using the reduced feature set. In general, we hypothesized that the hand-selected features would reclaim precision, hopefully without losing recall. For the most part, the initial results support this hypothesis. In comparison to the systems using all 53 features, we see statistically significant gains in precision and statistically significant, but much smaller, drops in recall, producing systems with better F-measure scores. (See Figure 6). In addition, precision on common nouns rises substantially, as expected (not shown in the figure).

5.6 Training Instance Selection

As noted before, training data for coreference resolution are typically created by relying on coreference chains from the training documents: training instances are generated by pairing each NP with each of its preceding NPs. Consequently, $\binom{n}{2}$ training instances will be generated from n NPs in a document in the absence of instance selection. Many learning-based coreference systems, including our Baseline system which implements the Soon et al. system, have one form of instance selection scheme or another, however. The need for instance selection stems from two intrinsic properties of coreference:

Coreference is a rare relation. That is, most NP pairs in a document are not coreferent. Consequently, generating training instances by pairing each NP with each of its preceding NPs creates highly skewed class distributions, in which the number of positive instances is overwhelmed by the number of negative instances. For example, the MUC-6 and MUC-7 coreference data sets used for dryrun contain only 2% positive instances. Unfortunately, learning in the presence of such skewed class distributions remains an open area of research in the machine learning community (e.g. Pazzani et al. (1994), Fawcett (1996), Cardie and Howe (1997), Kubat and Matwin (1997)).

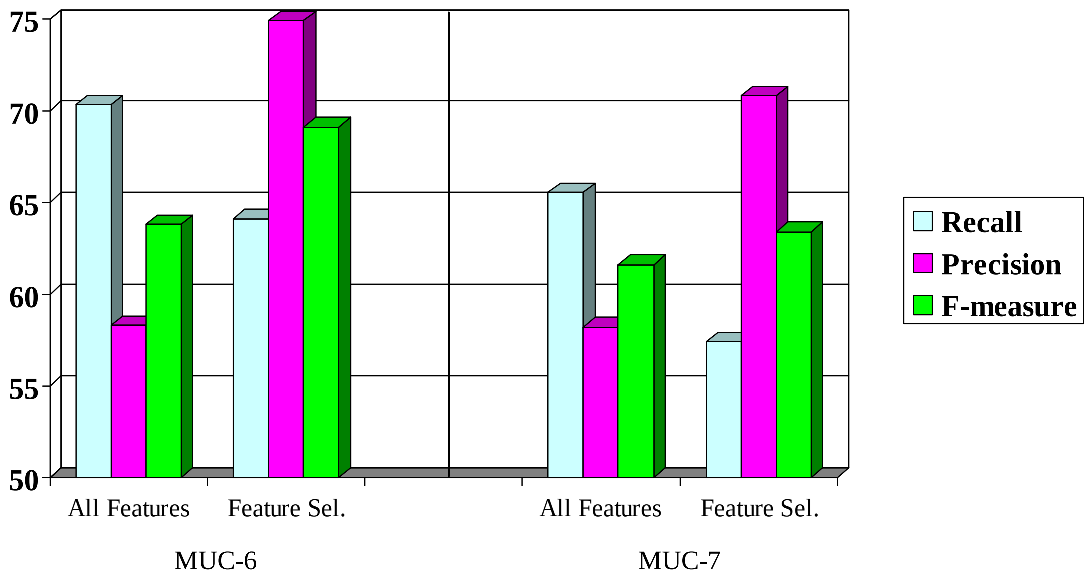

Figure 6: Effects of Manual Feature Selection

Coreference is a discourse-level problem with different solutions for different types of NPs. The interpretation of a pronoun, for example, may be dependent only on its closest antecedent and not on the rest of the members of the same coreference chain. Proper name resolution, on the other hand, may be better served by ignoring locality constraints altogether and relying on string-matching or more sophisticated aliasing techniques. Consequently, generating positive instances from all pairs of NPs from the same coreference chain can potentially make the learning task harder: all but a few coreference links derived from any chain might be hard to identify based on the available contextual cues.

While the first property calls for *negative instance selection*, the second one clearly demands *positive instance selection*. Previous work on instance selection for learning-based coreference systems can roughly be divided into two categories: class-independent selection (i.e. selection without supervisory class information) and class-dependent selection (i.e. selection that requires supervisory information). For example, Mueller et al. (2002) present a class-independent selection scheme for reducing the skewness of class distributions in the training set. Specifically, an antecedent-anaphor pair is discarded if one of the following conditions hold: (1) the anaphor is an indefinite NP; (2) the two NPs belong to different semantic classes; (3) one of the NPs is not a 3rd person noun phrase; (4) the two NPs do not agree in number or gender; (5) one NP spans (i.e. encloses) the other. Mueller et al. report that this method of instance selection halves the size of the original training set by removing mostly negative instances that they believe do not contribute any knowledge towards the classification. In addition, their system performs Soon et al.'s class-dependent selection method that we described in section 5.1.

Work to date and results. We have devised two class-dependent instance selection methods, one for negative instance selection and the other for positive instance selection (Ng and Cardie, 2002a). Our negative instance selection algorithm, NEG-SELECT, is similar to that proposed in Soon et al., but is designed to be used in conjunction with the best-first clustering algorithm that we described in section 5.4. In contrast to Soon et al.'s results, however, we show empirically that F-measure

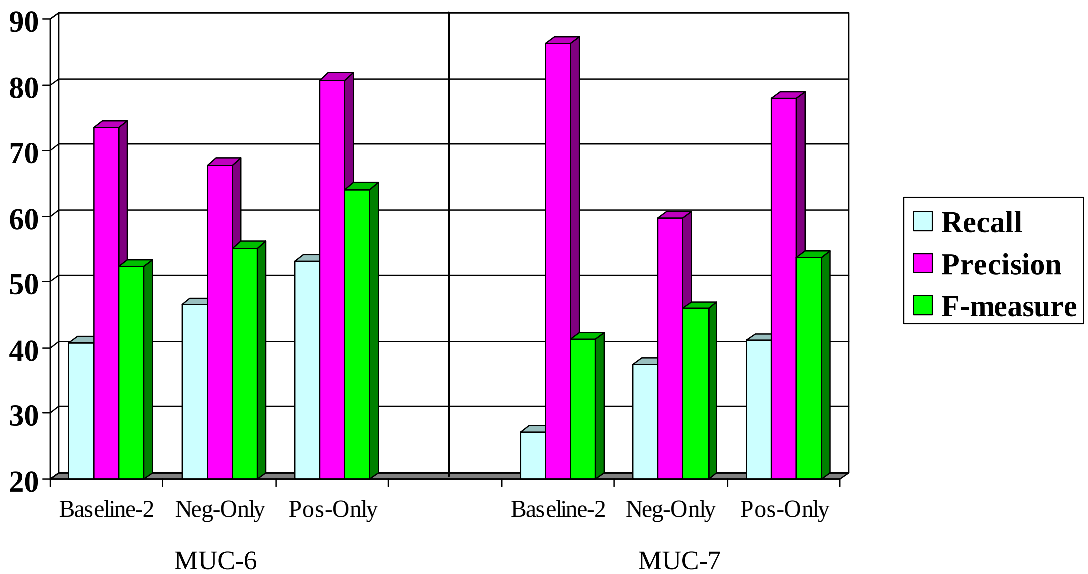

Figure 7: Effects of Training Instance Selection using NEG-SELECT and POS-SELECT

increases noticeably in response to negative instance selection. Figure 7 compares the results obtained using a baseline coreference system (Baseline-2) that employs no instance selection against the system that uses NEG-SELECT (Neg-Only). On the other hand, our positive instance selection algorithm, POS-SELECT, attempts to avoid the inclusion of "hard" training instances. While all existing instance selection methods for coreference resolution are heuristic-based, POS-SELECT is a corpus-based method for implicit selection of positive instances and is a fully automated variant of the instance selection algorithm introduced in Harabagiu et al. (2001). Although the corpus-based selection algorithm is designed to be used in conjunction with a rule-learning algorithm, the technique can in principle be used with any learning algorithm in which the learned hypothesis can be transformed into an equivalent set of rules (e.g. decision tree learners). With RIPPER as our underlying learning algorithm, experimental results indicate that F-measure increases substantially in response to positive instance selection. (See the Pos-Only results in Figure 7.)

5.7 Rule Selection

As noted above, in order to apply machine learning methods to coreference resolution, we need to recast the problem as a classification task. However, this conversion precludes the enforcement of the transitivity constraint. After training, for example, the classifier might determine that A is coreferent with B, and B with C, but that A and C are not coreferent. Hence, the clustering mechanism is needed to coordinate these possibly contradictory pairwise classifications. In addition, because the coreference classifiers are trained independently of the clustering algorithm to be used, improvements in classification accuracy do not guarantee corresponding improvements in clustering-level accuracy, i.e. overall performance on the coreference task might not improve. Unfortunately, none of the existing learning-based coreference systems attempt to deal with this problem.

Work to date. We have investigated a method that resolves this conflict by tightening classification- and clustering- level coreference decisions: we introduce an error-driven rule pruning algorithm that considers rules induced by the coreference classifier and discards those that cause the rule-

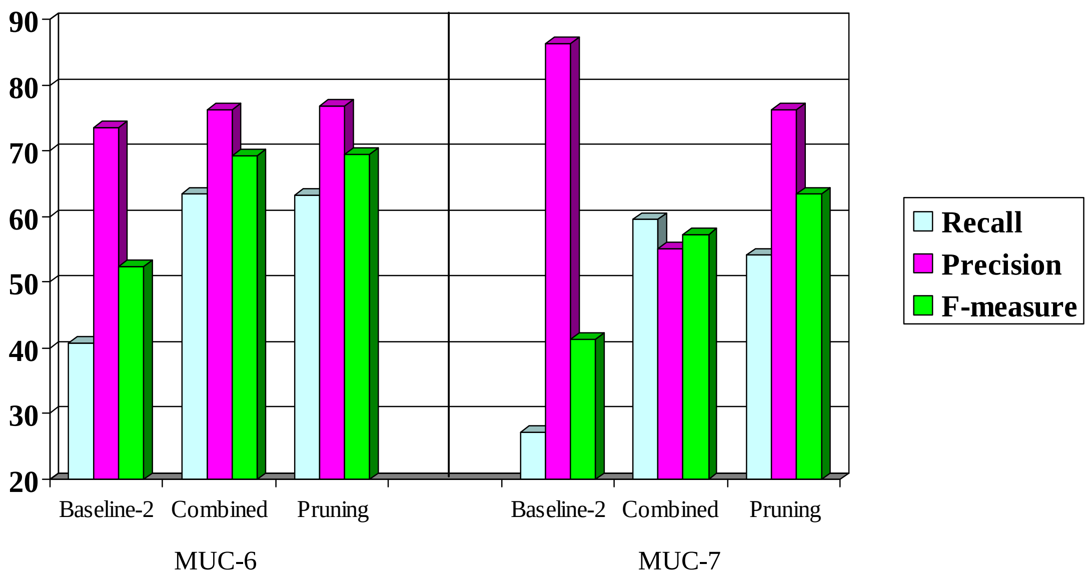

Figure 8: Effects of Rule Selection

set to perform poorly with respect to the global, clustering-level coreference scoring function (Ng and Cardie, 2002a). The algorithm, RULE-SELECT, takes as input a ruleset learned from a training corpus for performing coreference resolution, a pruning corpus (disjoint from the training corpus), and a clustering-level coreference scoring function that is the same as the one being used for evaluating the final output of the system.[17] At each iteration, RULE-SELECT greedily discards the rule whose removal yields a ruleset with which the coreference system performs the best (with respect to the coreference scoring function) on the pruning corpus. As a hill-climbing procedure, the algorithm terminates when removal of any of the rules in the ruleset fails to improve performance. In contrast to most existing algorithms for coreference resolution, RULE-SELECT establishes a tighter connection between the classification- and clustering- level decisions and ensures that system performance is optimized with respect to the coreference scoring function.

Results. Our experimental results have shown that RULE-SELECT, when applied in conjunction with NEG-SELECT and POS-SELECT, yields substantial improvements over a coreference system without instance selection. In Figure 8, the Combined results are obtained using a coreference system (Baseline-2) that employs both NEG-SELECT and POS-SELECT, whereas the Pruning results are obtained when NEG-SELECT, POS-SELECT, and RULE-SELECT are all applied to Baseline-2.

So far we have seen a number of directions in which improvements to Soon et al.'s basic machine learning framework can potentially be made. On the linguistic side, we have attempted to enhance the Soon feature set and make use of anaphoricity information for coreference resolution. On the machine learning side, we have proposed several precision-enhancing modifications to the Soon et al. learning framework and demonstrated that machine learning techniques such as feature selection, training instance selection, and rule selection can be used to improve system performance. We now conclude this section with our proposed work.

[17]Importantly, RULE-SELECT assumes *no* knowledge of the inner workings of the scoring function.

5.8 Proposed Research

In this subsection, we list the work we propose to do for the rest of the dissertation.

- As mentioned at the beginning of the section, different dimensions of improvement used slightly different baselines. In future work, we will incorporate the changes entailed by each dimension incrementally into a single baseline system (i.e. the Duplicated Soon Baseline). In addition, we will clean up the experimental method and perform additional analyses in order to better characterize the behavior of the coreference system and its various components.

- We plan to evaluate our approach on more data sets. One possibility is to use the formal evaluation data sets for training and test the resulting system on the dryrun data sets. Another corpus we are considering is the BLLIP corpus[18], which consists of three years of machine-parsed Wall Street Journal articles. Coreference information is annotated on top of the parse trees. We will look at whether this corpus is appropriate for evaluation purposes.

- We plan to automate the feature selection process, which currently employs a fair amount of user discretion, e.g. to determine a precision cut-off. Since wrapper approaches to feature selection (i.e. approaches that use the induction method as a subroutine) are generally more effective at selecting relevant features than filter approaches (i.e. approaches that filter out irrelevant features before induction occurs) (see Blum and Langley (1997)), we will start with two simple, but commonly used, wrapper methods for feature selection: forward selection and backward elimination. A forward selection algorithm starts with an empty set and greedily adds features one at a time until all attributes have been added. A backward elimination procedure starts with all features in the feature set and greedily removes them one at a time until no feature remains. While an exhaustive search for an optimal set of features would require an $O(2^n)$ time algorithm given n features, the two greedy algorithms run in time quadratic in the number of features. Indeed, both algorithms have been criticized for being too greedy (Caruana and Freitag, 1994). Consequently, Caruana and Freitag propose several new feature selection algorithms that reduce the degree of greediness by allowing the algorithms to add or remove features at any step of the search other than the first. We will consider these alternatives if they are deemed more appropriate.

- With regard to training instance selection, our current results have only shown that both NEG-SELECT and POS-SELECT can substantially improve the performance of a coreference system that does not employ any instance selection scheme. Future work involves replacing the heuristic-based instance selection component in the coreference resolution system described in section 5.5 with NEG-SELECT and POS-SELECT and comparing the effectiveness of heuristic-based and learning-based selection methods.

- With regard to rule selection, we plan to apply RULE-SELECT to the version of the coreference system using NEG-SELECT and POS-SELECT for instance selection together with the features chosen by the feature selection algorithm. In addition, we plan to develop rule-pruning algorithms that have a lower asymptotic time complexity than RULE-SELECT, which runs in time quadratic in the number of rules for a fixed pruning corpus and hence may become inefficient as the number of rules increases. We will probably proceed by making an even greedier search in the rule space than RULE-SELECT but still measuring the usefulness of a rule with respect to a held-out pruning corpus using the coreference scoring function as in RULE-SELECT.

[18]See `http://www.ldc.upenn.edu/Catalog/LDC2000T43.html` for more information.

6 Conclusion

Machine learning approaches to coreference resolution have been shown to be a promising way to build robust coreference resolution systems. Despite the successful application of machine learning techniques to coreference resolution, the problem is far from being solved. Our ultimate goal is to improve existing machine learning approaches to coreference resolution. In this paper, we have suggested several directions in which improvements can potentially be made. On the linguistic side, we have attempted to increase both the number and sophistication of the features available to the learning algorithm. We have also shown that automatically acquired anaphoricity information can be used to improve the precision of a coreference system. On the machine learning side, we have proposed several extra-linguistic modifications to the Soon et al. learning framework. Additionally, we have demonstrated that the system can be further optimized with such machine learning techniques as feature selection, training instance selection, and rule selection. Above all, many of these new components of the coreference system are learning-based. In contrast to most previous work that focuses on the automatic construction of coreference classifiers, we have made a step towards the automatic construction of a coreference resolution system in which all of its components can potentially be acquired automatically. With sufficient labeled data, it is conceivable that even the "best" learning algorithm and clustering algorithm for coreference resolution can be determined empirically.

Coreference resolution is undoubtedly one of the most difficult problems in language understanding. Given that NLP is often said to be "AI-complete"[19], we might be able to conclude that coreference resolution is among the hardest of the hardest problems in artificial intelligence. Nevertheless, as Mitkov (2001) puts it, the area is "difficult but not intractable", and recent advances in the problem have demonstrated that we are making "slow, but steady progress".

References

Hussein Almuallim and Thomas G. Dietterich. 1991. Learning with many irrelevant features. In *Proceedings of the Ninth National Conference on Artificial Intelligence*, pages 547–552, Anaheim, CA. AAAI Press / MIT Press.

Michael Anderberg. 1973. *Cluster Analysis for Applications*. Academic Press.

Chinatsu Aone and Scott William Bennett. 1995. Evaluating automated and manual acquisition of anaphora resolution strategies. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pages 122–129. Association for Computational Linguistics.

Saliha Azzam, Kevin Humphreys, and Robert Gaizauskas. 1998. Evaluating a focus-based approach to anaphora resolution. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and COLING-98*, pages 74–78. Association for Computational Linguistics.

Saliha Azzam, Kevin Humphreys, and Robert Gaizauskas. 1999. Using coreference chains for text summarization. In *Proceedings of the Workshop on Coreference and Its Applications*.

Saliha Azzam. 1996. Resolving anaphors in embedded sentences. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pages 263–268.

Amit Bagga and Breck Baldwin. 1998. Entity-based cross-document coreferencing using the vector

[19]This means that "the most difficult problems in artificial intelligence manifest themselves in human language phenomena" (see Lee (To appear)).

space model. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and COLING-98*, pages 79–85.

Breck Baldwin and Thomas Morton. 1998. Dynamic coreference-based summarization. In *Proceedings of the Third Conference on Empirical Methods in Natural Language Processing.*

B. Baldwin, M. Morton, A. Bagga, J. Baldridge, R. Chandraseker, A. Dimitriadis, K. Snyder, and M. Wolska. 1998. Description of the UPenn CAMP System as used for Coreference. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, San Francisco, CA. Morgan Kaufmann.

Breck Baldwin. 1997. CogNIAC: High precision coreference with limited knowledge and linguistic resources. In *Proceedings of the ACL'97/EACL'97 Workshop on Operational factors in practical, robust anaphora resolution*, pages 38–45.

David Bean and Ellen Riloff. 1999. Corpus-based identification of non-anaphoric noun phrases. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 373–380.

Avrim Blum and Pat Langley. 1997. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, pages 245–271.

Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT-98).*

Simon Botley and Anthony McEnery. 2000. Discourse Anaphora: The Need for Synthesis. In Simon Botley and Anthony McEnery, editors, *Corpus-based and Computational Approaches to Discourse Anaphora*, pages 1–41. John Benjamins Publishing Company.

E. Breck, J. Burger, L. Ferro, D. House, M. Light, and I. Mani. 1999. A sys called Qanda. In *Proceedings of the Eighth Text REtrieval Conference TREC 8.* NIST Special Publication.

Susan E. Brennan, Marilyn Walker Friedman, and Carl J. Pollard. 1987. A centering approach to pronouns. In *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics.* Association for Computational Linguistics.

Jaime Carbonell and Ralf Brown. 1988. Anaphora resolution: A multi-strategy approach. In *Proceedings of the 12th International Conference on Computational Linguistics*, pages 96–101.

Claire Cardie and Nicholas Howe. 1997. Improving minority class prediction using case-specific feature weights. In Doug Fisher, editor, *Proceedings of the Fourteenth International Conference on Machine Learning*, pages 57–65, Vanderbilt University, Memphis, TN. Morgan Kaufmann.

Claire Cardie and Kiri Wagstaff. 1999. Noun phrase coreference as clustering. In *Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-99)*, pages 82–89, University of Maryland, College Park, MD. Association for Computational Linguistics.

Claire Cardie. 1997. Empirical methods in information extraction. *AI Magazine*, 18(4):65–79.

Rich Caruana and Dayne Freitag. 1994. Greedy attribute selection. In *Proceedings of the Eleventh International Conference on Machine Learning*, San Francisco, CA. Morgan Kaufmann.

William Cohen. 1995. Fast effective rule induction. In *Proceedings of the Twelfth International Conference on Machine Learning*, San Francisco, CA. Morgan Kaufmann.

M. Craven, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and C.Y. Quek. 1997. Learning to extract symbolic knowledge from the world wide web. Internal report, School of Computer Science, CMU, January.

Arthur Dempster. 1968. A generalization of Bayesian inference. *Journal of the Royal Statistical Society*, 30:205–247.

Michel Denber. 1998. Automatic resolution of anaphora in English. Technical report, Eastman Kodak Co.

Richard Evans. 2001. Applying machine learning toward an automatic classification of it. *Literary and Linguistic Computing*, 16(1):45–57.

Tom Fawcett. 1996. *Learning with skewed class distributions — summary of responses.* Machine Learning List: Vol. 8, No. 20.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database.* MIT Press, Cambridge, MA.

R. Gaizauskas, T. Wakao, K. Humphreys, H. Cunningham, and Y. Wilks. 1995. Description of the LaSIE System as used for MUC-6. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, San Francisco, CA. Morgan Kaufmann.

Niyu Ge, John Hale, and Eugene Charniak. 1998. A statistical approach to anaphora resolution. In Charniak, Eugene, editor, *Proceedings of the Sixth Workshop on Very Large Corpora*, pages 161–170, Montreal, Canada. ACL SIGDAT.

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

Barbara J. Grosz, Aravind K. Joshi, and Scott Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–226.

Barbara J. Grosz. 1977. The representation and use of focus in a system for understanding dialogs. In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*, pages 67–76.

Liliane Haegeman. 1994. *Introduction to Government and Binding Theory.* Blackwell Publishers Ltd., 2nd edition.

Sanda Harabagiu, Răzvan Bunescu, and Steven Maiorano. 2001. Text and knowledge mining for coreference resolution. In *Proceedings of the Second Meeting of the North America Chapter of the Association for Computational Linguistics (NAACL-2001)*, pages 55–62.

Graeme Hirst. 1981. *Anaphora in Natural Language Understanding.* Springer Verlag.

Jerry Hobbs. 1978. Resolving pronoun references. *Lingua*, 44:311–338.

George John, Ron Kohavi, and Karl Pfleger. 1994. Irrelevant features and the subset selection problem. In W. Cohen and H. Hirsh, editors, *Proceedings of the Eleventh International Conference on Machine Learning*, pages 121–129, Rutgers University, New Brunswick, NJ. Morgan Kaufmann.

Megumi Kameyama. 1998. Intrasentential centering: A case study. In Marilyn Walker, Aravind Joshi, and Ellen Prince, editors, *Centering Theory in Discourse*, pages 89–112. Oxford University Press.

Andrew Kehler. 1997a. Current theories of centering for pronoun interpretation: A critical evaluation. *Computational Linguistics*, 23(3):467–475.

Andrew Kehler. 1997b. Probabilistic coreference in information extraction. In C. Cardie and R. Weischedel, editors, *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 163–173. Association for Computational Linguistics.

Christopher Kennedy and Branimir Boguraev. 1996. Anaphor for everyone: Pronominal anaphora resolution without a parser. In *Proceedings of the 16th International Conference on Computational Linguistics*, pages 113–118.

Miroslav Kubat and Stan Matwin. 1997. Addressing the curse of imbalanced training sets: One-Sided selection. In *Proceedings of the 14th International Conference on Machine Learning (ICML-97)*, pages 179–186. Morgan Kaufmann.

Shalom Lappin and Herbert Leass. 1994. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–562.

Lillian Lee. To appear. "I'm sorry Dave, I'm afraid I can't do that": Linguistics, Statistics and Natural Language Processing circa 2001. In *The National Research Council Study on Fundamentals of Computer Science*.

Joseph McCarthy and Wendy Lehnert. 1995. Using decision trees for coreference resolution. In C. Mellish, editor, *Proceedings of the Fourteenth International Conference on Artificial Intelligence*, pages 1050–1055.

Ruslan Mitkov, Branimir Boguraev, and Shalom Lappin. 2001. Introduction to the special issue on computational anaphora resolution. *Computational Linguistics*, 27(4):473–477.

Ruslan Mitkov, Richard Evans, and Constantin Orasan. 2002. A new, fully automatic version of Mitkov's knowledge-poor pronoun resolution method. In Al. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, pages 169–187. Springer.

Ruslan Mitkov. 1997. Factors in anaphora resolution: they are not the only things that matter. A case study based on two different approaches. In *Proceedings of the ACL'97/EACL'97 Workshop on Operational Factors in Practical, Robust Anaphora Resolution*.

Ruslan Mitkov. 1998. Robust pronoun resolution with limited knowledge. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 867–875.

Ruslan Mitkov. 1999. Anaphora resolution: The state of the art. Technical Report (Based on the ACL-COLING 98 tutorial on anaphora resolution), University of Wolverhampton, Wolverhampton.

Ruslan Mitkov. 2001. Outstanding Issues in Anaphora Resolution. In Al. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, pages 110–125. Springer.

Thomas Morton. 2000. Coreference for NLP applications. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*.

MUC-6. 1995. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*. Morgan Kaufmann, San Francisco, CA.

MUC-7. 1998. *Proceedings of the Seventh Message Understanding Conference (MUC-7)*. Morgan Kaufmann, San Francisco, CA.

Christoph Mueller, Stefan Rapp, and Michael Strube. 2002. Applying co-training to reference resolution. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 352–359.

Ion Muslea, Steve Minton, and Craig Knoblock. 2002. Active + Semi-Supervised Learning = Robust Multi-View Learning. In *Proceedings of the Nineteenth International Conference on Machine Learning*.

Vincent Ng and Claire Cardie. 2002a. Combining sample selection and error-driven pruning for machine learning of coreference rules. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 55–62.

Vincent Ng and Claire Cardie. 2002b. Identifying anaphoric and non-anaphoric noun phrases to improve coreference resolution. In *Proceedings of the 19th International Conference on Computational Linguistics*.

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

C. Paice and G. Husk. 1987. Towards the automatic recognition of anaphoric features in English text: the impersonal pronoun 'it'. *Computer Speech and Language*, 2.

M. Pazzani, C. Merz, P. Murphy, K. Ali, T. Hume, and C. Brunk. 1994. Reducing misclassification costs. In *Proceedings of the Eleventh International Conference on Machine Learning*, pages 217–225, Rutgers University, New Brunswick, NJ. Morgan Kaufmann.

Ellen Prince. 1981. Toward a Taxonomy of Given-New Information. In P. Cole, editor, *Radical Pragmatics*, pages 223–255. New York, N.Y.: Academic Press.

J. Ross Quinlan. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA.

G. Salton, A. Wong, and C.S. Yang. 1975. A vector space model for information retrieval. *Communications of the ACM*, 18(11):613–620.

Candace Sidner. 1979. *Towards a Computational Theory of Definite Anaphora Comprehension in English Discourse*. PhD Thesis, Massachusetts Institute of Technology.

Candace Sidner. 1981. Focusing for interpretation of pronouns. *American Journal of Computational Linguistics*, 7:217–231.

Amit Singhal, Steve Abney, Michiel Bacchiani, Michael Collins, Donald Hindle, and Fernando Pereira. 1999. AT&T at TREC-8. In *Proceedings of the Eighth Text REtrieval Conference TREC 8*. NIST Special Publication.

Stephen Soderland. 1997. Learning to extract text-based information from the World Wide Web. In *Proceedings of Third International Conference on Knowledge Discovery and Data Mining (KDD-97)*, pages 251–254. AAAI Press.

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

Michael Strube. 1998. Never look back: An alternative to centering. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 1251–1257.

Joel Tetreault. 1999. Analysis of syntax-based pronoun resolution methods. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 602–605.

Kees van Deemter and Rodger Kibble. 2000. On coreferring: Coreference in MUC and related annotation schemes. *Computational Linguistics*, 26(4).

Renata Vieira and Massimo Poesio. 2000. Processing Definite Descriptions in Corpora. In S. Botley and A. McEnery, editors, *Corpus-based and Computational Approaches to Discourse Anaphora*. UCL Press.

M. Vilain, J. Burger, J. Aberdeen, D. Connolly, and L. Hirschman. 1995. A model-theoretic coreference scoring scheme. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pages 45–52, San Francisco, CA. Morgan Kaufmann.

Marilyn Walker. 1989. Evaluating discourse processing algorithms. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, pages 251–261.