Universiteit Antwerpen
Faculteit Letteren en Wijsbegeerte

Optimization Issues in Machine Learning of Coreference Resolution

---

Optimalisatie van Lerende Technieken
voor Coreferentieresolutie

Proefschrift voorgelegd tot het behalen van de graad van
doctor in de Taal- en letterkunde
aan de Universiteit Antwerpen te verdedigen door
Véronique HOSTE

Promotor: Prof. Dr. W. Daelemans Antwerpen, 2005

Cover based on painting "T,JI" from Günter Tuzina

Scientific knowledge always remains sheer guesswork - although guesswork controlled by criticism and experiment.
-Karl Popper
-Realism and the Aim of Science, p. 13

If you knew some of the experiments (if they may be so-called) which I am trying, you would have a good right to sneer, for they are so absurd even in my opinion that I dare not tell you.
-Charles Darwin
-Letter to J.D. Hooker, April 14th, 1855

Abstract

This thesis presents a machine learning approach to the resolution of coreferential relations between nominal constituents in Dutch. It is the first automatic resolution approach proposed for this language. The corpus-based strategy was enabled by the annotation of a substantial corpus (ca. 12,500 noun phrases) of Dutch news magazine text with coreferential links for pronominal, proper noun and common noun coreferences. Based on the hypothesis that different types of information sources contribute to a correct resolution of different types of coreferential links, we propose a modular approach in which a separate module is trained per NP type. Lacking comparative results for Dutch, we also perform all experiments for the English MUC-6 and MUC-7 data sets, which are widely used for evaluation.

Applied to the task at hand, we focus on the methodological issues which arise when performing a machine learning of language experiment. In order to determine the effect of algorithm 'bias' on learning coreference resolution, we evaluate the performance of two learning approaches which provide extremes of the eagerness dimension, namely TIMBL as an instance of lazy learning and RIPPER as an instance of eager learning. We show that apart from the algorithm bias, many other factors potentially play a role in the outcome of a comparative machine learning experiment. In this thesis, we study the effect of selection of information sources, parameter optimization and the effect of sampling to cope with the skewed class distribution in the data. In addition, we investigate the interaction of these factors.

In a set of feature selection experiments using backward elimination and bidirectional hillclimbing, we show the large effect feature selection can have on classifier performance. We also observe that the feature selection considered to be optimal for one learner cannot be generalized to the other learner. Furthermore, in the parameter optimization or model selection experiments, we observe that the performance differences within one learning method are much larger than the method-comparing performance differences. A similar observation is made in the experiments exploring the interaction between feature selection and parameter optimization, using a genetic algorithm as a computationally feasible way to achieve this type of costly optimization. These experiments also show that the parameter settings and information sources which are selected after optimization cannot be generalized. In the experiments varying the class distribution of the training data, we show that both learning approaches behave quite differently in case of skewedness of the classes and that they also react differently to a change in class distribution. A change of class distribution is primarily beneficial for RIPPER. However, we observe that once again no particular class distribution is optimal for all data sets, which makes this resampling also subject to optimization.

In all optimization experiments, we show that changing any of the architectural variables can have great effects on the performance of a machine learning method, making questionable conclusions in the literature based on the exploration of only a few points in the space of possible experiments for the algorithms to be compared. We show that there is a high risk that other areas in the experimental search space lead to radically different results and conclusions.

At the end of the thesis, we move away from the instance level and concentrate on the coreferential chains reporting results on the Dutch and English data sets. In order to gain an insight into the errors committed in the resolution, we perform a qualitative error analysis on a selection of English and Dutch texts.

Samenvatting

Dit proefschrift gaat over het gebruik van lerende technieken voor de resolutie van coreferentiële relaties tussen nominale constituenten in het Nederlands. Het is meteen de eerste automatische aanpak voor deze taal. Die corpusgebaseerde aanpak werd mogelijk gemaakt door de annotatie van een aanzienlijk corpus van teksten uit een Vlaams weekblad met nieuws uit de nationale en internationale actualiteit. Tijdens de annotatie werden ongeveer 12,500 nominale constituenten, bestaande uit eigennamen, soortnamen en pronomina, voorzien van coreferentiële informatie. Uitgaande van de hypothese dat het type informatie dat nodig is voor een correcte resolutie kan verschillen per type coreferentiële relatie, hebben we gekozen voor een modulaire aanpak waarbij een aparte module getraind wordt voor elk type van nominale constituent. Aangezien er nog geen vergelijkbare resultaten beschikbaar zijn voor het Nederlands hebben we onze experimenten ook uitgevoerd en geëvalueerd op de Engelse MUC-6 en MUC-7 data sets.

Toegepast op de taak van coreferentieresolutie gaan we dieper in op de methodologische aspecten die meespelen bij de toepassing van lerende systemen op natuurlijke taal. In een eerste reeks experimenten gaan we het effect na van de zogenaamde 'bias', de zoekheuristieken die een bepaalde leertechniek gebruikt en de manier waarop de geleerde kennis over de uit te voeren taak gerepresenteerd wordt. Daartoe evalueren we de performantie van twee lerende technieken die kunnen beschouwd worden als twee extremen in het continuum van lerende systemen, namelijk het geheugengebaseerde systeem TIMBL en het regelinductie-

systeem RIPPER. We tonen aan dat naast de bias van het algoritme nog veel andere factoren potentieel een rol spelen in het uiteindelijke resultaat van een leerexperiment. In dit proefschrift bestuderen we het effect van de selectie van informatiebronnen, van de optimalisatie van de parameters en het effect van sampling op datasets met scheefgetrokken klassedistributies. Verder gaan we de interactie na tussen deze factoren.

In een reeks experimenten waarbij op een automatische manier relevante kennisbronnen (features) geselecteerd worden, tonen we het grote effect van featureselectie aan op de performantie van het leersysteem. We observeren ook dat de optimale featureselectie voor een bepaalde leertechniek niet kan veralgemeend worden naar andere leertechnieken. In een reeks experimenten waarbij de algoritmeparameters systematisch gevarieerd worden, tonen we verder nog aan dat de performantieverschillen binnen eenzelfde leertechniek veel groter kunnen zijn dan de performantieverschillen tussen twee of meerdere leertechnieken. Een gelijkaardige observatie kunnen we ook maken in de experimenten waarbij gekeken wordt naar de interactie tussen featureselectie en parameteroptimalisatie. Om dit soort rekenintensieve optimalisatie mogelijk te maken, wordt gebruik gemaak van een genetisch algoritme. Deze experimenten geven ook aan dat de parameterinstellingen en de kennisbronnen die geselecteerd worden na optimalisatie niet kunnen gegeneraliseerd worden. In de experimenten waarbij de klassedistributie van de data gevarieerd wordt, tonen we aan dat beide leertechnieken zich verschillend gedragen bij scheefgetrokken klassen en dat ze ook verschillend reageren op een verandering in die distributie. Een verandering in de klasseverdeling blijkt vooral gunstig voor RIPPER. Maar ook hier kunnen we geen welbepaalde distributie aanduiden die optimaal is voor alle datasets.

Alle optimalisatie-experimenten tonen aan dat een wijziging in een van de architecturale variabelen een groot effect kan hebben op de performantie van een leermethode. Door deze conclusie komen bestaande conclusies in de literatuur op de helling te staan, omdat die vaak gebaseerd zijn op het exploreren van maar enkele punten in de experimentele ruimte. Onze studie toont aan dat er een groot risico bestaat dat andere plaatsen in de experimentele zoekruimte tot radicaal verschillende resultaten en conclusies kunnen leiden.

Op het einde van het proefschrift verlaten we het instantieniveau en concentreren we ons op de coreferentiële kettingen door de resultaten te rapporteren op de Nederlandse en Engelse testcorpora. Met het oog op een beter begrip van de fouten die begaan zijn tijdens de resolutie hebben we een kwalitatieve foutenanalyse doorgevoerd op een selectie van Engelse en Nederlandse teksten.

Acknowledgments

I must confess. One of the things I am not good at, is saying "thank you". Gratitude is sometimes difficult if everything in life goes rather smoothly. I would like to seize this opportunity to express my gratitude.

This dissertation could not have been initiated nor completed without the help and support of my supervisor, Walter Daelemans. During the years I worked here, Walter has become much more for me than "just" my thesis supervisor. I would like to thank him for his faith in me and for creating a stimulating research environment. I am also grateful for his guidance and his encouragement in moments of dwindling self-esteem. I really look forward to our continued cooperation. Credit is also due to Steven Gillis and Marie-Laure Reinberger for the valuable comments they made during the preparation of this dissertation. I would like to thank Claire Cardie and Antal van den Bosch for agreeing to be in my thesis committee and for the time and effort they spent in reviewing this dissertation. A special thanks is due to Walter De Mulder for reading and commenting on the introductory chapter.

I also want to acknowledge the current and past members of the CNTS research group for the support, the coffee breaks, for bringing me a sandwich every noon and for the pleasant working atmosphere - my roommate Anja, Guy, Hanne, Helena, Anne, Masja, Marie-Laure, Gert D., Gert V.R., Griet, Fien, Erik, Evie, Bart, Kim, Jo, Eric, Karen, Frederik, Kevin. I thank the FWO for funding the PROSIT project I worked on during the last four years. I am also grateful that my colleagues in the project Erwin, Martin, Antal and Walter have granted me

plenty of freedom to pursue my research on coreference resolution.

I would also like to thank the people from the CNTS and ILK research groups who have created the whole machinery of learning tools used among others for preprocessing. My special thanks go to Erik Tjong Kim Sang, with whom I shared a room for more than 6 years, for creating and adapting the shallow parser and more importantly, for his always listening ear and valuable feedback. I would also like to acknowledge Bart Naudts for creating and maintaining the GA software on the cluster.

I also have the pleasure of always being able to count on friends, whether it was for a laugh or a tear.

I am also one of the fortunate people that was brought up well, with a father and mother that loved me and always tried to do the best for me. I thank them for giving me such a positive foundation upon which to grow. I am especially grateful to my mother for her patience and belief in me in moments of fear of failure. I have become very attached to her, sometimes therapeutic, phone calls when driving to Antwerp. I thank my little sister Charlotte for being a precious sister and friend. I thank Jan among other things for his creativity in making the cover for this dissertation. A warm thanks also goes to my family in law for their love and support.

Thank you, Christophe, my husband and dearest friend, for your love and for being such a challenging companion in life! Thanks, Emile, our wonderful son, just for being there!

Lo vidi, e'l primo palpito
il cor sentì d'amore;
mi vide appena, e il core
balzò del mio fedel.
Quaggiù si riconobbero
nostr'alme in rincontrarsi
formate per amarsi
Iddio le avea in ciel!
- Luisa Miller, Giuseppe Verdi

Contents

1 Introduction **1**

1.1 Defining coreference 2

1.2 The task of coreference resolution 3

1.3 Research objectives 8

1.3.1 Automatic coreference resolution for Dutch 9

1.3.2 Methodological issues in machine learning of coreference resolution 9

1.4 Dissertation outline 10

2 Coreferentially annotated corpora **13**

2.1 Coreference annotation 13

2.2 MUC-6 and MUC-7 16

2.2.1 Annotation markup 16

2.2.2 Annotated relations 18

2.2.3 Resulting data sets 21
2.3 KNACK-2002 21
2.3.1 Annotation markup 21
2.3.2 Annotated relations 22
2.3.3 Resulting data sets 26
2.4 Inter-annotator agreement 26
2.5 Summary 27

3 Information sources **29**
3.1 Data preparation 30
3.1.1 Preprocessing 30
3.1.2 Positive and negative instances 34
3.1.3 One vs. three 37
3.2 Selection of informative features 40
3.2.1 The choice of features in related work 40
3.2.2 Our shallow information sources 49
3.2.3 The informativeness of the features in a feature vector . . 56
3.3 Summary 57

4 Machine learning of coreference resolution **61**
4.1 The 'bias' of the machine learner 61
4.2 TIMBL, a lazy learner 64
4.3 RIPPER, a greedy learner 68
4.4 Baseline experiments 71
4.4.1 Experimental setup 71
4.4.2 Evaluation measures 72

4.4.3 Results on the validation data 74

4.5 Summary 80

5 Selecting the optimal information sources and algorithm settings 83

5.1 There is more to it than 'bias' 84

5.2 Feature selection 86

5.2.1 Filters and wrappers 87

5.2.2 Feature informativeness 88

5.3 Searching the feature space 89

5.3.1 Backward elimination 91

5.3.2 Bidirectional hillclimbing 92

5.4 Variation of algorithm parameters 99

5.5 Summary: the need for combined optimization 104

6 Genetic algorithms for optimization 105

6.1 Genetic algorithms 106

6.2 A genetic algorithm approach for feature selection and parameter optimization 109

6.2.1 Experimental setup 109

6.2.2 Experimental results 111

6.3 The optimal features and parameter settings 114

6.4 Summary and discussion 118

7 The problem of imbalanced data sets 123

7.1 Learning from imbalanced data sets 124

7.2 Machine learning research on imbalanced data sets 125

7.2.1 Sampling 126
7.2.2 Adjusting misclassification costs 127
7.2.3 Weighting of examples 128
7.3 Imbalanced data sets in coreference resolution 129
7.3.1 Instance selection in the machine learning of coreference resolution literature 130
7.3.2 Investigating the effect of skewedness on classifier performance 131
7.4 Balancing the data set 133
7.4.1 Random 134
7.4.2 Exploiting the confusion matrix 135
7.5 Summary and discussion 140

8 Testing **145**

8.1 Data preparation 146
8.1.1 Search scope 146
8.1.2 Resulting data sets 147
8.2 Shift from the instance level to the coreference chain level 149
8.2.1 Antecedent selection 151
8.2.2 Evaluation procedure 152
8.3 Experimental results 155
8.3.1 Classifier results 156
8.4 Error analysis 158
8.4.1 MUC-7 158
8.4.2 KNACK-2002 165
8.5 Summary 171

9 Conclusion **173**

9.1 Methodological issues: main observations 174

9.2 Future research goals 180

References **181**

A Manual for the annotation of coreferences in Dutch newspaper texts **197**

A.1 Introduction 197

A.1.1 Coreference and anaphora 198

A.1.2 Types of coreference 200

A.1.3 Encoding coreferential relations 201

A.2 Annotation scheme 203

A.2.1 Names and named entities 204

A.2.2 Pronouns 206

A.2.3 Conjoined noun phrases 208

A.2.4 NPs containing relative clauses 209

A.2.5 Other phrases without a head noun 210

A.3 Special cases 211

A.3.1 Bound anaphors 211

A.3.2 "Paycheck" pronouns 212

A.3.3 Appositions 212

A.3.4 Predicate nominals 215

A.3.5 Time-dependent identity 216

A.3.6 Metonymy 218

A.3.7 Set relations, possessive relations, discontinuous NPs, etc. 219

A.4 The coreference attributes 219

A.5 The ALEMBIC Workbench 220
A.5.1 Starting the Alembic Workbench 220
A.5.2 5 Menus 221
A.6 How to annotate? 223
A.6.1 Annotation procedure 223
A.6.2 Selecting phrases 223
A.6.3 Different coreference tags 223
A.7 An example annotation 224

B Ripper rules for the MUC-6 "Proper nouns" data set 227

C Three MUC-7 documents for which a qualitative error analysis has been carried out 231

D Three KNACK-2002 documents for which a qualitative error analysis has been carried out 243

CHAPTER 1

Introduction

This thesis is about the automatic resolution of coreference using machine learning techniques. It is a research area which is becoming increasingly popular in natural language processing (NLP) research and it is a key task in applications such as machine translation, automatic summarization and information extraction for which text understanding is of crucial importance. When people communicate, they aim for cohesion. Text is therefore "not just a string of sentences. It is not simply a large grammatical unit, something of the same kind as a sentence, but differing from it in size–a sort of supersentence, a semantic unit."(Halliday and Hasan 1976, p. 291). Coreference, in which the interpretation of an element in conversation depends on a previously mentioned element, is one possible technique to achieve this cohesion, a technique to construct that supersentence. Through the use of shorter or alternative linguistic structures which refer to previously mentioned elements in spoken or written text, coherent communication can be achieved. A good text understanding largely depends on the correct resolution of these coreferential relations.

In this introductory chapter, we provide a definition of coreference and anaphora (Section 1.1) and discuss existing knowledge-based and corpus-based approaches to the task of automatic coreference resolution (Section 1.2). The remainder of the chapter introduces the present study, lists the central research objectives

(Section 1.3) and gives an overview of this thesis (Section 1.4).

1.1 Defining coreference

In the literature, no unequivocal definition on coreference and anaphora can be found. According to Hirst (1981),

> anaphora is the device of making in discourse an abbreviated reference to some entity (or entities) in the expectation that the perceiver of the discourse will be able to disabbreviate the reference and thereby determine the identity of the entity. The reference is called an anaphor, and the entity to which it refers is its referent or antecedent. A reference and its referent are said to be coreferential. The process of determining the referent of an anaphor is called resolution.(p. 4-5)

A more narrow definition which focuses on the differences between coreferences and anaphors has been proposed by Kibble (2000) and van Deemter and Kibble (2000). They provide two textbook definitions from Trask (1983) to provide a clear view on these differences. They consider **coreference** as the relation which holds between two NPs (e.g. NP_1 and NP_2) both of which are interpreted as referring to the same extralinguistic entity (Referent(NP)). In short: NP_1 and NP_2 corefer if and only if Referent(NP_1) = Referent(NP_2). This means that a coreference relation is an *equivalence relation*. Furthermore, coreferential relations are *symmetrical* (if NP_1 and NP_2 corefer, this implies that also NP_2 and NP_1 corefer) and also *transitive* (if NP_1 and NP_2 corefer and if also NP_2 and NP_3 corefer, this implies that also NP_1 and NP_3 will corefer). This transitivity can alleviate the task of coreference resolution, as suggested in McCarthy (1996). But transitivity also implies that wrongly assigned coreference relations will cause even more errors. Another feature of coreference relations is that there is *no context sensitivity of interpretation*. Note that Kibble (2000) and van Deemter and Kibble (2000) interpret coreference as reference to the same extralinguistic entity, to something in the real world. However, reference can also point to non-existing entities, to entities evoked in discourse (Karttunen 1976). Also according to Kibble (2000) and van Deemter and Kibble (2000), an **anaphor** is an item (e.g. NP_1) with little intrinsic meaning or reference which takes its interpretation from another item (e.g. NP_2) in the same sentence or discourse, its antecedent. In other words, NP_1 takes NP_2 as its anaphoric antecedent if and only if NP_1 depends on NP_2 for its interpretation. As opposed to the coreference relation, the anaphoric relation is

nonsymmetrical. It also implies *context sensitivity of interpretation*. An expression is anaphoric only if it depends for its interpretation on a contextually given item.

The simplest discourse anaphors are coreferential. If anaphoric and coreferential relations coincide, they denote one discourse referent. Anaphoric relations involving definite pronouns and definite noun phrases are mostly coreferential. However, not all anaphoric relations are coreferential, which implies that the anaphor in that case does not denote the same discourse referent as the antecedent, but rather a discourse referent which is associated with a discourse referent given in discourse. These non-coreferential anaphors exist under different names, viz. indirect anaphora, partial anaphora, associative anaphora, bridging anaphora, etc. (see for example Clark (1975) and Fraurud (1992)). Examples of such associative definite anaphors are "The bus came around the corner. **The driver** was drunk.", or "If the gas tank is empty, you should refuel **the car**".

The description above might give the impression that the discussion on coreferential/anaphoric relations is restricted to noun phrases. In the past decades, however, there have been several proposals to also interpret other phenomena as anaphora, such as tense (as in Partee (1973) and Webber (1998), e.g. "Sam and Emma had a car accident. Emma **got hurt**."), VP ellipsis (as in Hardt (1992), e.g. "Mike loves to party. Sam does too."), etc.

In this thesis, the focus is on coreferential relations between noun phrases. Although the discussion whether or not a given referring link between two nominal constituents can be qualified as coreferential, anaphoric or both is beyond the scope of this thesis, we will return to this issue in the next chapter, discussing the annotation of the MUC-6 and MUC-7 corpora and the annotation decisions made for the KNACK-2002 corpus.

1.2 The task of coreference resolution

As stated before, a good text understanding also depends on the correct resolution of the coreferential relations it contains. The resolution of coreferential relations is a complex task since it requires finding the correct antecedent among many possibilities. As shown in the two examples listed below, it involves different types of knowledge: morphological and lexical knowledge such as number agreement and knowledge about the type of noun phrase, syntactic knowledge such as information about the syntactic function of anaphor and antecedent, semantic knowledge which allows us to recognize synonyms and hyperonyms or which allows distinctions to be made between person, organization or location

names, discourse knowledge, world knowledge, etc. These information sources must enable an automatic resolution system to resolve that "we" are not "fruit flies" and that the "they" in the second example refers to "the kidnappers", presuming that Alfred Heineken did not consider 43 million guilders as being "modest".

> Fruitvliegen zijn om vele redenen succesvoller dan mensen. En **wij** helpen **hen** daarenboven nog een handje met dingen die **zij** dan weer niet kunnen. (KNACK-2002 training data)
> English: Fruit flies are for many reasons more successful than humans. And **we** help **them** with the things **they** are not able to do.

> Op 9 november 1983 werd Alfred Heineken samen met zijn chauffeur ontvoerd. **De kidnappers** vroegen 43 miljoen gulden losgeld. Een bescheiden bedrag, vonden **ze** zelf. (KNACK-2002 training data)
> English: On 9 November 1983 Alfred Heineken and his driver were kidnapped. **The kidnappers** asked a ransom of 43 million guilders. A modest sum, **they** thought.

In the research on computational coreference resolution, different directions can be taken. We will now discuss some of these existing approaches, thereby focusing on the approaches aiming at the unrestricted resolution of nominal coreferences (pronouns, proper noun NPs and common noun NPs) in English. For a more elaborate overview on the literature on anaphora resolution from its early days to more recent work, we refer to Mitkov (2002).

Among **the knowledge-based approaches** to coreference resolution, a distinction can be made between approaches which generally depend upon linguistic knowledge, as in Hobbs (1978), and the discourse-oriented approaches, in which discourse structure is taken into account, as in Grosz, Joshi and Weinstein (1995). In these approaches, there has been an evolution from systems requiring an extensive amount of linguistic and non-linguistic knowledge (e.g. Rich and LuperFoy (1988)) toward more knowledge-poor approaches (e.g. Mitkov (1998)). Furthermore, there has been a shift from the more theoretically oriented approaches (not or hardly performing any system evaluation) toward more practical working systems.

The systems depending on *linguistic knowledge* (lexical, morphological, syntactic, semantic) for the resolution of coreferential relations apply this linguistic knowledge through the use of *constraints and preferences*. Lexical, morphological, syntactic, and semantic knowledge is used to define these constraints

and preferences. Whereas the constraints are applied in order to remove bad antecedents, the preferences impose an ordering on the remaining candidate antecedents.

One of the early approaches to coreference resolution which is still popular, is Hobbs's approach (Hobbs 1978). This approach implies a surface parse tree which builds up a search space of the sentence containing the anaphor and the preceding sentences, which is searched in a left-to-right, breadth-first manner. A match is found when the antecedent NP in question matches in gender, number and person with the anaphoric pronoun. Hobbs also uses selectional restrictions to rule out bad candidate antecedents. Lappin and Leass (1994) describe an algorithm for pronominal anaphora resolution which is primarily based on the syntactic information present in a full syntactic parse-tree of the text. The system uses a syntactic filter on pronoun-NP coreference, a procedure for identifying pleonastic pronouns, different salience parameters (such as grammatical role, parallelism of grammatical roles, frequency of mention, proximity and sentence recency), etc.

Due to the high error rate in case of full syntactic parsing, several alternatives to full parsing have been proposed ranging from partial parsing (e.g. Palomar, Ferrández, Moreno, Martìnez-Barco, Peral, Saiz-Noeda and Muñoz (2001) and Kennedy and Boguraev (1996)) to part-of-speech tagging (e.g. Baldwin (1997) and Mitkov (1998)). Kennedy and Boguraev (1996), for example, modify the Lappin and Leass algorithm in a way that it works on a flat syntactic analysis (provided by a part-of-speech tagger and a noun phrase grammar) of the text. An alternative to this approach was proposed by Stuckardt (2001), whose ROSANA system operates on the deficient output of a syntactic parser. Another well-known system is CogNIAC from Baldwin (1997) for the resolution of gendered pronouns. The CogNIAC system starts from a part-of-speech tagged and base-NP chunked corpus and uses a limited set of anaphora resolution rules, which are applied in a predefined order. Mitkov (1998) is another example of a system making use of part-of-speech information. The system identifies the noun phrases in a context of two sentences and checks the candidate antecedents for gender and number agreement and a number of antecedent indicators (such as definiteness, givenness, lexical reiteration, distance, etc.).

Discourse information has also been used for automatic anaphora resolution. Especially centering (Grosz, Joshi and Weinstein 1983, Grosz et al. 1995, Walker, Joshi and Prince 1998, Poesio, Stevenson, di Eugenio and Hitzeman 2004) and focusing theory (Sidner 1979) have been successfully used. Both theories start from the assumption that certain entities mentioned in an utterance are more central/in focus than others and this imposes certain constraints on the referential relations in a text. There are two basic types of centers, namely a so-called backward looking center and the so-called forward looking centers,

which are a set of discourse entities ranked according to their salience. Through the application of constraints, possible antecedents are filtered out because of morphosyntactic, binding and semantic criteria. The BFP algorithm of Brennan, Friedman and Pollard (1987) and the work of Tetreault (2001), Strube and Hahn (1999) and Hardt (2004) are examples of systems using the centering framework.

Several systems try to combine different theories on anaphora resolution. The work of Rich and LuperFoy (1988), for example, combines the principles of Discourse Representation Theory (Kamp 1981), centering, focusing, etc. in different modules. Each module proposes candidate antecedents and evaluates other module's proposals. The shallow processing approach from Carter (1987) and the multi-strategy approach from Carbonell and Brown (1988) are other examples of systems combining different information sources.

In the last decades, **corpus-based techniques** have become increasingly popular for the resolution of coreferential relations. Whereas corpus-based techniques have become the norm for many other natural language processing tasks (such as part-of-speech tagging, parsing, grapheme-to-phoneme conversion, etc.), the field of computational coreference resolution is still highly knowledge-based. The corpus-based approach starts from the assumption that the knowledge necessary for the correct resolution of coreferential relations is present in the annotated data. The use of corpus-based techniques was enabled by the creation of coreferentially annotated corpora, such as MUC-6 and MUC-7. Dagan and Itai (1990), for example, derive *collocation patterns* from corpora and use these patterns to filter out unlikely antecedent candidates. Another corpus-based approach is proposed in the COCKTAIL system of Harabagiu, Bunescu and Maiorano (2001), in which coreference rules are mined from coreferentially annotated corpora. COCKTAIL covers both nominal and pronominal coreferences and uses different sets of heuristics for the different types of anaphors. The heuristics are learned from tagged corpora and are applied in a predefined order. Ge, Hale and Charniak (1998), use a *statistical approach* for the resolution of third person anaphoric pronouns. The algorithm has two components. One component collects the statistics of the training corpus (part of Penn Tree-bank text (Marcus, Santorini and Marcinkiewicz 1993)) and the other uses the probabilities based on the training material to resolve pronouns in the test corpus. During training, they run the Hobbs algorithm on the Tree-bank parse trees until it has proposed n candidates for each pronoun. As information sources they consider the distance between the pronoun and the candidate antecedent, information on number, gender and animacy of the antecedent, etc. and calculate statistics for this training information. During testing, they again consider n candidates and the probabilities collected on the training set are applied to each of these candidates. The one with the highest combined probability is selected as the

antecedent.

Also *machine learning techniques* have gained popularity in the research on coreference resolution. In a machine learning approach to coreference resolution, information on pairs of NPs is represented in a set of feature vectors. These vectors most often contain distance, morphological, lexical, syntactic and semantic information on the candidate anaphor, its candidate antecedent and also on the relation between both. The goal of this feature information is to enable the machine learner to distinguish between coreferential and non-coreferential relations. The machine learning techniques can be divided into two groups, depending on the information they receive.

In case of *unsupervised learning*, the learner receives no feedback on the coreferentiality of two noun phrases. Cardie and Wagstaff (1999), for example, view coreference resolution as a clustering task which combines noun phrases into equivalence classes. Each noun phrase in the input text is represented as a set of 11 features (e.g. the individual words, the head noun, the position of the NPs, the pronoun type, information on (in)definiteness, gender, number, animacy and semantic class) and a distance metric is calculated between each set of 2 noun phrases. The clustering algorithm then considers the distance between noun phrases and if this distance is less than the clustering radius r, their classes are considered for possible merging or clustering. Two coreference classes can be merged if they do not contain incompatible NPs.

Most machine learning approaches to coreference resolution, however, are *supervised techniques*, learners which receive feedback on whether a given pair of NPs is coreferential or not. Examples of such systems are the C4.5 decision tree learner (Quinlan 1993) as used by Aone and Bennett (1995), McCarthy (1996), Soon, Ng and Lim (2001), Yang, Zhou, Su and Tan (2003) and Yang, Su, Zhou and Tan (2004a), maximum entropy learning as in Kehler (1997) and Luo, Ittycheriah, Jing, Kambhatla and Roukos (2004) and the Ripper rule learner (Cohen 1995) as in Ng and Cardie (2002a), Ng and Cardie (2002b) and Ng and Cardie (2002c). All these approaches recast the problem as a classification task: a classifier is trained to decide whether a pair of NPs is coreferential or not. The pair of NPs is represented by a feature vector containing information on both NPs and the relation between them. Instances are labeled positive if the NPs are coreferential and negative if they are not. The methods experiment with different types of features (morphological, syntactic, semantic, etc.) and with different sizes of feature vectors (varying between 53, as in Ng and Cardie (2002c) and 8 features, as in McCarthy and Lehnert (1995)). In Chapter 3, we will discuss in more detail the different information sources which can be relevant for coreference resolution.

Hybrid approaches in which corpus-based techniques are incorporated in a knowledge-based system have been used for coreference resolution as well. Hartrumpf (2001), for example, combines syntactico-semantic rules with statistical knowledge derived from an annotated corpus. Dagan, Justeson, Lappin, Leass and Ribak (1995) use statistically measured collocation patterns to rank antecedent candidates, and Byron and Allen (1999), Orasan, Evans and Mitkov (2000) and Preiss (2002) use machine learning techniques (genetic algorithms and memory-based learning, respectively) to determine the weights of the different information sources.

1.3 Research objectives

The work reported in this thesis falls within the framework of the corpus-based approaches, and more specifically the machine learning approaches to coreference resolution. There are several reasons for this choice in favor of a machine learning approach. First, this type of approach was enabled by the availability of different shallow information sources, such as part-of-speech information, NP chunk information, information about named entities, and so on. Furthermore, in contrast to the knowledge-based techniques, a machine learning method allows us to empirically determine which information sources contribute to the correct resolution of coreferential links, without making use of linguistic presuppositions. Another motivation to use a machine learning strategy for coreference resolution is that this type of approach has shown to outperform state-of-the-art knowledge-based approaches (for example on the MUC-6 and MUC-7 data).

One of the distinguishing features of this study is that it is the first significant automatic approach to the resolution of coreferential relations between nominal constituents in Dutch. The second distinguishing feature, compared to the research described in the previous section, is the extensive optimization in the classification phase. Other methods pay few attention to this first classification step and report mainly about the second step in which one antecedent is selected per anaphor. Our study focuses on the optimization issues which arise when performing a machine learning of language experiment and for the first time applies these methodological insights to the problem of coreference resolution for both English and Dutch.

We will now discuss these two objectives in more detail.

1.3.1 Automatic coreference resolution for Dutch

This thesis presents a machine learning approach to the resolution of coreferential relations between nominal constituents in Dutch. It is the first corpus-based resolution approach proposed for this language. Beside the fact that not much research has been done yet on automatic Dutch coreference resolution, the existing research on this topic from op den Akker, Hospers, Lie, Kroezen and Nijholt (2002) and Bouma (2003) falls within the knowledge-based resolution framework and only focuses on the resolution of pronominal anaphors.

The corpus-based strategy was enabled by the annotation of a new corpus with coreferential relations between noun phrases. This annotation effort was crucial since the existing corpora for Dutch only contain anaphoric relations for pronouns and are rather small. The annotated corpus of op den Akker et al. (2002), for example, consists of a small number of texts from different types (newspaper articles, magazine articles and fragments from books) and only contains 801 annotated pronouns. Another small corpus for Dutch was annotated by Bouma (2003). It is based on the Volkskrant newspaper and contains anaphoric relations for 222 pronouns. In Chapter 2, we discuss the annotation of a substantial Dutch corpus of coreferential relations between about 12,500 noun phrases, including named entities, definite and indefinite NPs and pronouns. The corpus is based on KNACK, a Flemish weekly news magazine with articles on national and international current affairs, covering a wide variety of topics in economical, political, scientific, cultural and social news.

Not only did the annotation effort enable us to assess the difficulty of the task, it also led to a corpus which can be used for the evaluation and the development of different approaches to automatic coreference resolution for Dutch.

Based on the hypothesis that different types of information sources contribute to a correct resolution of different types of coreferential links, we propose a modular learning approach in which a separate module is trained per NP type. This implies that a separate module is trained for the pronominal coreferences, the proper noun coreferences and the common noun coreferences. Lacking comparative results for Dutch, we apply the same type of approach to the English MUC-6 and MUC-7 data sets, which are widely used for evaluation.

1.3.2 Methodological issues in machine learning of coreference resolution

During the machine learning experiments, different methodological issues arise concerning a methodologically sound and transparent experimental design. Do

the results of a given machine learning method on the task of coreference resolution method only depend on the algorithm 'bias' of the specific learner? Or do other factors play a role in the outcome of a comparative machine learning experiment? We hypothesize that the use of a methodology in which only a few points in the experimental search space are searched, can lead results which are totally different than the ones obtained when searching other areas in this space.

In order to test this hypothesis, the following subquestions are investigated:

- What is the effect of algorithm 'bias' on learning coreference resolution? In order to study this effect, we evaluate the performance of two learning approaches which provide extremes of the eagerness dimension, namely RIPPER (Cohen 1995) as an instance of eager learning and TIMBL (Daelemans, Zavrel, van der Sloot and van den Bosch 2002) as an instance of lazy learning.
- Does feature selection have an effect on classifier performance? Can the information sources considered to be optimal for one learner be generalized to other learning methods?
- What is the effect of parameter optimization on classifier performance? Can the optimal parameters for a given learning method be generalized to the different data sets?
- How does the class distribution of the training data affect learning? Are both learning approaches sensitive to a skewedness of the classes and do they react differently to a change in class distribution?
- What is the effect of the interaction of all these factors?

The answers to these research questions are not only interesting for anyone working on machine learning of coreference resolution. This discussion about experimental methodology also applies to every machine learning (of language) experiment.

1.4 Dissertation outline

This thesis is structured as follows. Chapter 2 introduces the annotated corpora for Dutch and English which will be used in all experiments in this thesis. It also discusses the annotation schemes used for the annotation of these corpora.

Chapter 3 deals with the problem of the selection of information sources for the resolution of coreferential relations. It first discusses the preparation of the data sets, including the preprocessing of the data, the selection of positive and negative instances, and the construction of three different data sets instead of one single data set, namely one for the pronominal noun phrases, one for the named entities and one for the other noun phrases. It also reviews the different shallow information sources which have been used in other machine learning work on coreference resolution. It continues with a description of the positional, contextual, morphological, lexical, syntactic and semantic features that are used in the experiments in this thesis.

Chapter 4 introduces the term 'bias' and the two machine learning packages which are used in the experiments: the memory-based learning package TIMBL (Daelemans et al. 2002) and the rule induction package RIPPER (Cohen 1995). The second part of this chapter provides a description of the general setup of the experiments, of the different classifier performance measures and it applies the two learning methods to the coreference resolution data sets. The results show that the precision scores for TIMBL are much lower than the ones for RIPPER, whereas the opposite tendency is observed with respect to the recall scores.

Chapter 5 is the first of three chapters discussing methodological issues when performing a machine learning of language experiment. The chapter considers the importance of feature selection and the importance of the optimization of algorithm parameters. It systematically shows that changing any of the architectural variables can have great effects on the performance of a learning method.

In Chapter 6, we proceed to a next optimization step in a set of experiments exploring the interaction between feature selection and parameter optimization. Given the combinatorially explosive character of this joint optimization, a genetic algorithm is used in conjunction with our two learning methods TIMBL and RIPPER. The results in this and the previous chapter show that the variability recorded for the same algorithm when doing feature selection, parameter optimization and their joint optimization is often much larger than the difference between the two learning algorithms. At the end of the chapter, we broaden this conclusion to other data sets.

Chapter 7 adds yet another dimension to this discussion on factors influencing a machine learning experiment: the effect of class distribution on classifier performance. The first sections of the chapter discuss several strategies for dealing with skewed class distributions. The remainder of the chapter concentrates on the sensitivity of both TIMBL and RIPPER to class imbalances in our data sets and on their sensitivity to rebalancing the class distributions. We show that both learning approaches behave quite differently and explain this different be-

havior by the nature of the learning approaches. Each of the three previous chapters shows the importance of optimization, a topic which is often neglected in the machine learning of language research. The experiments in all three chapters show that exploring only a few points in the experimental search space is a questionable methodology and that there is a high risk that other areas in this space may lead to radically different results and conclusions.

In Chapter 8, we leave the discussion on methodology and return to the task at hand: coreference resolution. This chapter moves away from the instance level and concentrates on the coreferential chains. The chapter first describes the new experimental setup and the new evaluation procedure. And it then reports the results of TIMBL and RIPPER on the different data sets. The chapter concludes with a qualitative error analysis.

Chapter 9 summarizes the conclusions of this thesis and suggests future work.

CHAPTER 2

Coreferentially annotated corpora

In the experiments reported in this thesis, we use two inductive learning methods, viz. memory-based learning and rule induction, to resolve coreferential relations between nominal constituents. Since these corpus-based methods depend on the quality of the corpora they are trained on, we will discuss in this chapter the importance of coreference annotation. Section 2.1 introduces the topic of coreference annotation. In Section 2.2 and Section 2.3, we introduce the two corpora we will use for our experiments: the well-known and widely used MUC-6 and MUC-7 corpora for English and the newly developed KNACK-2002 corpus for Dutch. Section 2.2 describes the MUC-6 and MUC-7 annotation markup, the annotated relations and the resulting training and test corpora. Section 2.3 has a similar setup but focuses on the distinctive features of the KNACK-2002 annotation guidelines. Section 2.4 discusses the problem of inter-annotator agreement.

2.1 Coreference annotation

The annotation of corpora with coreferential information is useful from both a linguistic and a computational point of view. From a linguistic perspective,

coreferentially annotated corpora provide insight in the frequency of different types of coreferences, the type of relations between them, etc. From a computational perspective, these corpora can be used for the development and evaluation of automatically trained systems. This type of approach has already been used by Aone and Bennett (1995), Fisher, Soderland, Mccarthy, Feng and Lehnert (1995), McCarthy and Lehnert (1995), McCarthy (1996), Ge et al. (1998), Cardie and Wagstaff (1999), Soon et al. (2001), Strube, Rapp and Müller (2002), Ng and Cardie (2002a, 2002b, 2002c) and others. Coreferentially annotated corpora can also be used for evaluation of knowledge-based coreference resolution systems.

There are however not many corpora available that are annotated with coreferential links and that are publicly available. In this thesis, we will perform experiments on the English data from the MUC (Message Understanding Conferences) coreference task and on a new Dutch coreferentially annotated corpus. The MUC data (MUC-6 and MUC-7) are chosen for the experiments since they are widely used for training and evaluating coreference resolution systems. For Dutch, a new corpus is developed since there is as far as we know up to now no corpus available in which coreferential relations are encoded between noun phrases.

For the annotation of the MUC data, and also for the annotation of the Dutch data, a **coreference annotation scheme** was developed to guide the annotation with coreferential information. Two main decisions have to be taken in such an annotation scheme:

- It has to be decided between *what type of constituents* coreferential relations are encoded. These coreference relations can be established between different types of constituents, such as clauses (Passoneau 1996, Passoneau and Litman 1997, Fligelstone 1990), pronouns, noun phrases (MUC-6 1995, MUC-7 1998, Davies, Poesio, Bruneseaux and Romary 1998, Poesio and Vieira 1998), etc. The annotation scheme of MUC-6 (MUC-6 1995) and MUC-7 (MUC-7 1998) and the annotation scheme for the Dutch data (Appendix A) only covers coreferential relations between pronouns and noun phrases.
- Having determined for which type of constituents coreference annotation will be provided, it is decided *what type of coreferential relations* will be annotated. There are many different types of coreferential relations which can be encoded for noun phrases (see for example Webber (1978), McCarthy (1996)):
  - Identity relations as in **Xavier Malisse** *has qualified for the semi-finals in Wimbledon.* **The Flemish tennis player** *will play*

against an unknown opponent. In the previous example, there is an identity relation between *"Xavier Malisse"* and *"The Flemish tennis player"*.

- Type/token relations as in *I prefer* **the red car***, but my husband wanted* **the grey one**. In the example sentence, we are talking about two distinct cars, a red one and a grey one. *"The grey one"* denotes something like an object type rather than an object token. Also pronouns can enter into this type of relations, e.g. *Mark spent* **his first paycheck***, but Evelyn put* **it** *on her account.* Such a pronoun is known in literature as a 'paycheck pronoun' (see for example Cooper (1979), Gardent (2000)). Hirst (1981) uses the term "ISA", identity of sense anaphora, to denote this type of relationship in which the anaphor does not refer to the same entity as its antecedent but to one of a similar description.
- Part-whole/ element-set relations, e.g. *If* **the gas tank** *is empty, then the recommended action is to refuel* **the car**, or *Bill found himself in the middle of* **a forest**. **The trees** *were tall and sturdy.* This type of coreferential relations is also called "associative anaphora" (see for example Hawkins (1978), Bunescu (2003)).
- Nominal ellipsis as for example commonly used in recipes, e.g *Roll out* **bottom pie crust** *and place ϕ in 10 inch pie pan and set ϕ aside.* In this example, the arguments of "place" and "set aside" have been omitted.
- (...)

The annotation scheme of MUC-6 (MUC-6 1995) and MUC-7 (MUC-7 1998) and the core scheme from Davies et al. (1998) only cover the identity relation. They do not cover other types of coreference relations, such as set/subset, part/whole or type/token relations. Other schemes, such as the extended scheme from Davies et al. (1998) and the ones from Fligelstone (1990), Passoneau (1996), Passoneau and Litman (1997), Tutin, Trouilleux, Clouzot, Gaussier, Zaenen, Rayot and Antoniadis (2000) and others encode more relations.

Our experiments are performed on two coreferentially annotated datasets: MUC-6/MUC-7 and KNACK-2002. The experiments on the English MUC-6/MUC-7 data allow us to compare our approach with the work of Cardie and Wagstaff (1999), Soon et al. (2001), Ng and Cardie (2002a,2002b,2002c) and others. Since the KNACK-2002 corpus is built for this thesis, no comparative results are available yet. However, the corpus will be made publicly available and we hope that it will be used in the near future for the development and evaluation of coreference resolution systems for Dutch. We will now continue with a more detailed

description of the MUC-6 and MUC-7 data (Section 2.2) and the KNACK-2002 data (Section 2.3).

2.2 MUC-6 and MUC-7

MUC-6 and MUC-7 were two conferences in a series of Message Understanding Conferences (MUC). In the conferences preceding MUC-6 and MUC-7, the focus was mainly on the evaluation of information extraction systems. In MUC-6 and MUC-7, this focus was broadened and two new tasks were introduced: a named entity task involving the recognition of people, organization and place names, temporal expressions, etc. and a coreference task involving the identification of coreferential relations between noun phrases.

The MUC-6 and MUC-7 datasets are both based on newspaper articles: Wall Street Journal articles were used for the development of the MUC-6 coreference data and the MUC-7 coreference data are based on New York Times articles. The MUC-6 texts mainly contain economical news; the MUC-7 training data are mainly about airlines and plane crashes. The MUC data sets, however, contain a rather limited number of words and there is still need for much more annotation efforts, as for example those described in Orasan (2000). Other more recent important data sets for coreference resolution are the ACE (Automatic Content Extraction) data sets[1], which provide more annotated data, but which are based on other guidelines and use a different evaluation procedure.

2.2.1 Annotation markup

In the MUC annotations, coreferential links are marked between an antecedent and an anaphor. All coreferences start with a <COREF> tag and are closed with a </COREF> close tag. As illustrated in example (1) the markup of all antecedents contains one obligatory attribute ("ID") and two optional attributes ("MIN" and "STATUS"). The annotation of the coreferring NP contains two additional attributes, "TYPE" and "REF". We will now briefly discuss these different attributes.

- The "ID" is a unique ID given to the NP.
- The "MIN" string will in general be the head of the phrase. It indicates the minimum string that the system evaluated must include. The "MIN"

[1] See http://www.itl.nist.gov/iaui/894.01/tests/ace/index.htm for more information on these data sets.

(indicated in boldface in the following examples) attribute includes:

- the main noun without its left and right modifiers and in headless constructions the last token of the NP preceding any prepositional phrases, relative clauses and other 'right' modifiers, e.g. *the Clinton* **administration**,
- in the case of names, the entire name marked without any personal titles or modifiers, e.g. *Chairman* **John Dingell**,
- in the case of conjunctions in which the two components are annotated as one NP, the minimal phrase starting at the head of the first conjunct and including everything up to the end of the head for the last conjunct, e.g. **Eileen Cook and her 22-month-old daughter**, *Jessie*

If the maximal NP consists of a single head or a head preceded by an article, the "MIN" need not be marked. Furthermore, no "MIN" attribute is marked for dates, currency amounts and percentages.

- The "STATUS" attribute is used to indicate optional antecedents, as for example predicate relations or ISA relations as in *"Several years ago it merged its helicopter operations with* **those** *of Daimler Benz"*. Since the annotation of these relations falls outside the scope of the annotation of the identity relations, they are considered optional and errors on these relations are not taken into account in the evaluation software.

- The "TYPE" attribute is in fact superfluous, since only one type of coreference relation is marked, viz. the identity relation ("IDENT").

- The "REF" attribute indicates that there is a coreference between two noun phrases. The "REF" attribute links the current NP referring to a previously mentioned NP. A sequence of noun phrases referring to each other is called a "coreference chain". An example of such a chain is *"Michael D. Casey , a top Johnson & Johnson manager - its president - chief operating officer"* in example (1).

(1) <COREF ID = "0" TYPE = "IDENT" REF = "1" MIN = "Michael D. Casey"> Michael D. Casey , a top <COREF ID = "2" TYPE = "IDENT" REF = "3"> Johnson & Johnson </COREF> manager ,</COREF> moved to <COREF ID = "4" TYPE = "IDENT" REF = "5" MIN = "Genetic Therapy Inc."> Genetic Therapy Inc. , a small biotechnology concern <COREF ID = "6" TYPE = "IDENT" REF="7"> here </COREF> , </COREF> to become <COREF ID = "9" TYPE = "IDENT" REF = "0" MIN = "president" STATUS = "OPT"> <COREF ID = "8" TYPE = "IDENT" REF = "4"> its

</COREF> president </COREF> and <COREF ID = "10" TYPE = "IDENT" REF = "0" MIN = "officer" STATUS = "OPT"> chief operating officer </COREF> .

2.2.2 Annotated relations

The MUC annotation schemes (MUC-6 1995) and (MUC-7 1998) present **a set of guidelines** for marking up coreferences between noun phrases. We will now briefly discuss these guidelines:

- **Names and named entities** can enter into coreference relations: names of companies, organizations, persons, locations, dates, times, currency amounts, percentages, etc. Substrings of named entities are not marked. Dates are marked as a whole.

  (2) One reason **Lockheed Martin Corp.** did not announce a full acquisition of **Loral Corp.** on Monday, according to Bernard Schwartz, **Loral**'s chairman, was that **Lockheed** could not meet the price he had placed on **Loral**'s 31 percent ownership of **Globalstar Telecommunications Ltd. Globalstar** plans to provide telephone service by bouncing signals off 48 low-orbiting satellites.

- Furthermore, **personal (3), demonstrative (4), possessive (5) and reflexive pronouns (6)** can all enter into coreference relations. In the cases where pronouns have no antecedent at all or where they refer to something beyond the scope of annotation, such as a clausal construction, no coreference is marked.

  (3) (...) according to **Bernard Schwartz**, Loral's chairman, was that Lockheed could not meet the price **he** had placed (...)

  (4) Telepiu Robert Hersov said it will be **the first digital satellite television in Europe**. "**This** is a revolution in television," he said .

  (5) This deal means that **Bernard Schwartz** can focus most of **his** time on Globalstar

  (6) **Loral Space and Globalstar's 10 other partners** will put up the money **themselves**.

- When **2 or more noun phrases are conjoined or disjoined**, the MUC-6 and MUC-7 annotation schemes give different guidelines. According to the MUC-6 annotation scheme, noun phrases which contain 2 or more

heads are not annotated. The MUC-7 annotation scheme, however, states that it may be necessary to mark up the larger noun phrase (in italics in the example below) as well as the constituent noun phrases, depending on whether it is referred to later in the dialogue. For the annotation of the Dutch texts, we followed the MUC-7 guidelines.

(7) **Ms. Washington** and **Mr. Dingell** have been considered allies of the securities exchanges. (MUC-6)

(8) That 's certainly how **Eileen Cook** *and* **her 22-month-old daughter***, Jessie ,* see it. (MUC-7)

- Phrases with **nominalized adjectives, infinitives, gerunds or quantifiers** as heads can also enter into coreference relations.

(9) In **New York Stock Exchange composite trading** yesterday, it was quoted at 42.125, up 87.5 cents a share.

- All **predicate nominals** can enter into coreference relations. Davies et al. (1998), however, claim that predicative noun phrases (often indefinite noun phrases) cannot be considered to refer. This approach of integrating predicate nominals into coreference relations has also been criticized by van Deemter and Kibble (2000). In the annotation itself, this guideline of annotating all predicate nominals has not been strictly followed. Predicative indefinite noun phrases are rarely annotated as coreferential. For Dutch, the same strategy was used as the one proposed in the MUC annotation guidelines.

(10) **Dean** is **the nation's second-largest dairy, behind Borden Inc.**

(11) She currently is a counsel to the committee. (no annotation)

- For the annotation of **time-dependent identities**, two noun phrases are recorded as coreferential if the text asserts them to be coreferential disregarding different points in time. This implies that in (12) there will be a coreference chain between *"Barry Diller"*, *"its chairman and chief executive officer"*, *"Mr. Diller"* and *"chairman and chief executive of Fox Inc. and Twentieth Century Fox Film Corp., both units of News Corp"*. Problematic in this MUC-approach (see also van Deemter and Kibble (2000)) is that coreference is agreed to be an equivalence relation. This implies that in (12) *"its chairman and chief executive officer"* and *"chairman and chief executive of Fox Inc. and Twentieth Century Fox Film Corp."* can be used interchangeably. This is clearly not the case.

(12) QVC Network Inc., as expected, named **Barry Diller its chairman and chief executive officer**. (...) **Mr. Diller** previously was **chairman and chief executive of Fox Inc. and Twentieth Century Fox Film Corp.**, both units of News Corp. (MUC-6 test set)

- For the annotation of **functions and values**, the MUC annotation scheme stipulates that only **the most recent value** can corefer with the function. The other values are put into a separate coreference class.

(13) (...) he was pleased with **Sun's gross margins for the quarter**, which were **39%**, up sharply from 23% a year earlier

- The MUC annotation scheme stipulates to mark a coreference relation between a **bound anaphor** and the NP which binds it, as in (14).

(14) Nevertheless, **one union official** said **he** was intrigued by the brief and polite letter, which was hand-delivered by corporate security officers to the unions. (MUC-6 training data)

This approach has been criticized by van Deemter and Kibble (2000). They state that

$$\mathrm{NP}_1 \text{ and } \mathrm{NP}_2 \text{ corefer if and only if } \mathrm{Referent}(\mathrm{NP}_1) = \mathrm{Referent}(\mathrm{NP}_2).$$

- For the annotation of **appositions**, the MUC manual proposes to tag the noun phrase as a whole as well as any separate noun phrase contained in the appositive clause, if the appositive clause is contiguous to the noun phrase. The appositions refer to the complete noun phrase. Also indefinite appositions are marked. Furthermore, according to the MUC guidelines, appositional phrases are not marked when they are negative or when there is only partial overlap of sets.

(15) (...) is expected to nominate **Samuel Sessions**, **the committee's chief tax counsel**

- Coreference relations are also marked between **metonyms**, (e.g. "monarch" and "crown")

(16) We read where **the Clinton White House** is seeking a deputy to chief of staff Mack McLarty. (...) Are we supposed to conclude from this that it's OK to let "diversity " put women in high visibility jobs, but that when **the administration** needs to actually get something done and crack heads, the job has to go to a male pol? (MUC-6 test data)

2.2.3 Resulting data sets

For both MUC-6 and MUC-7, thirty documents annotated with coreference information were used as training documents. The MUC-6 and MUC-7 train set contain 1644 and 1905 anaphoric NPs, respectively. The test set for MUC-6 contains thirty documents and 1627 anaphoric noun phrases. Twenty texts, containing 1311 anaphoric NPs, serve as MUC-7 test set.

2.3 KNACK-2002

For Dutch, a new corpus was developed since there is as far as we know up to now no corpus available in which the coreferential relations are encoded between noun phrases. The existing corpora for Dutch only contain anaphoric relations for pronouns and are rather small. The annotated corpus of op den Akker et al. (2002), for example, consists of a small number of texts from different types (newspaper articles, magazine articles and fragments from books) and only contains 801 annotated pronouns. Bouma (2003) annotated a small corpus from the Volkskrant newspaper with 222 pronouns.

Lacking a substantial Dutch corpus of anaphoric relations between different types of noun phrases, including named entities, definite and indefinite NPs and pronouns, we annotated a corpus ourselves. Our Dutch coreferentially annotated corpus is based on KNACK, a Flemish weekly news magazine with articles on national and international current affairs. KNACK covers a wide variety of topics in economical, political, scientific, cultural and social news. For the construction of this Dutch corpus, we used a selection of articles of different lengths from KNACK, which all appeared in the first ten weeks of 2002. The corpus consists of 267 documents annotated with coreference information. In this corpus, 12,546 noun phrases are annotated with coreferential information.

2.3.1 Annotation markup

For the development of the annotation scheme (Appendix A) for our Dutch corpus, we took the MUC-7 (MUC-7 1998) manual and the manual from Davies et al. (1998) as source. In the annotation manual from Davies et al. (1998), the MUC coreference scheme is used as core scheme and an extended scheme also contains additional types of coreference. We also took into account the critical remarks from Kibble (2000) and van Deemter and Kibble (2000). For the annotation of the coreference relations in the KNACK-2002 corpus, we used

MITRE's "Alembic Workbench" as annotation environment[2].

As the MUC-6 (MUC-6 1995) and MUC-7 corpora (MUC-7 1998), the KNACK-2002 corpus was annotated with coreference links between noun phrases. The complete annotation manual for the Dutch annotators is provided in Appendix A.

(17) Ongeveer een maand geleden stuurde < COREF ID = "1" > American Airlines < /COREF > < COREF ID = "2" MIN = "toplui" > enkele toplui < /COREF > naar Brussel. < COREF ID = "3" TYPE = "IDENT" REF = "1" MIN="vliegtuigmaatschappij" > De grote vliegtuigmaatschappij < /COREF > had interesse voor DAT en wou daarover < COREF ID = "5" > de eerste minister < /COREF > spreken. Maar < COREF ID = "6" TYPE = "IDENT" REF = "5" > Guy Verhofstadt < /COREF > (VLD) weigerde < COREF ID = "7" TYPE = "BOUND" REF = "2" > de delegatie < /COREF > te ontvangen.

English: About one month ago, American Airlines sent some senior executives to Brussels. The large airplane company was interested in DAT and wanted to discuss the matter with the prime minister. But Guy Verhofstadt (VLD) refused to see the delegation.

In (17), three coreference chains (sequences of NPs referring to each other) are marked: one for *"American Airlines"* and *"De grote vliegtuigmaatschappij"*, a second chain with *"enkele toplui"* and *"de delegatie"* and a third chain with *"de eerste minister"* and *"Guy Verhofstadt"*. The annotation of this example sentence and all other sentences in our Dutch corpus mainly follows the MUC-7 guidelines (MUC-7 1998). As in the MUC annotations, all coreferences start with a <COREF> tag and are closed with a </COREF> close tag. The initial <COREF> tag contains additional information about the coreference: the unique ID of the NP (ID), the type of coreference relation (TYPE), the ID of the entity referred to (REF) and optionally the minimal tag of the coreference (MIN) and a "TIME" attribute.

2.3.2 Annotated relations

We will now focus on the differences between the MUC and the KNACK annotations. Whereas the MUC annotation scheme only describes one type coreference relation, viz. the identity relation ("IDENT"), we also mark other types of coreference relations, namely "BOUND", "ISA" and "MOD". We will now discuss

[2]More information on this workbench can be found at http://www.mitre.org/tech/alembic-workbench.

these three types and then continue with some other distinctive features of the KNACK annotation guidelines.

- As in the MUC annotations, we also marked a coreference relation between **a bound anaphor** and the NP which binds it, as in (18). Taking into account the critical remarks from Kibble (2000) and van Deemter and Kibble (2000) that we cannot consider this type of relation as an identity relation (see Section 2.2), we defined a new type of relation (as also proposed by Davies et al. 1998): "BOUND".

  (18) **Geen enkele Argentijn** kan meer dan 1100 euro per maand van **zijn** rekening halen.

  English: **No Argentine** can withdraw more than 1100 euro per month from **his** bank account.

- Frequently, anaphors (such as the above described "paycheck pronouns") do not refer to the same referent as their respective antecedents, as in (19). In this example sentence, there is no identity relation between the antecedent noun phrase *"time credit contributions"* and the referring noun phrase *"those of the federal government"*. In order to capture this type of relationships, we follow the definition of Hirst (1981) and distinguish between **identity of sense anaphora (ISA)** and **identity of reference anaphora (IRA)**. An IRA (in the MUC and in our annotation scheme: "IDENT") is an anaphor which denotes the same entity as its antecedent. An ISA anaphor does not denote the same entity as its antecedent, but one of a similar description.

  (19) Enkele dagen eerder immers had de Waalse regering de voet op de institutionele rem gezet om een einde te maken aan **de tijdskredietpremies** die de Vlaamse regering betaalt bovenop **die van de federale overheid**.

  English: A couple of days before the Walloon government put a break on further splitting up the institutions in order to end **the so-called "time credit" contributions** which are paid by the Flemish government on top of **those of the federal government**.

- We did also record coreference when the coreferential relation between two noun phrases is marked as possible rather than effective, as in (20). This type of coreferential relations is marked with the "MOD" attribute. The main motivation to annotate this type of relations is that they can also be informative in an information extraction task.

(20) **Schiphol**, **tot op heden de meest waarschijnlijke overnemer van BIAC**, heeft zijn bod ingetrokken.

English: **Schiphol**, **until now the most likely candidate for taking over BIAC**, has withdrawn its bid.

- For the annotation of **time-dependent identities**, which indicate a change over time, we followed the MUC-approach, also taking into account the criticism from van Deemter and Kibble (2000) and Davies et al. (1998). We added a time-indication in the annotation of these noun phrases (expressed in the "TIME" attribute). Also for the annotation of **functions and values**, this "TIME" attribute was used. An example of the use of this "TIME" attribute is given in (21), in which "55" is marked with "TIME=1" and "58 jaar" with "TIME=2".

  (21) Minister van Onderwijs Marleen Vanderpoorten (VLD) wil **de uitstapleeftijd** optrekken van **55** naar **58 jaar**.

  English: The Secretary of Education Marleen Vanderpoorten (VLD) intends to raise **the retiring age** from **55** to **58 years**.

- Finally, for the annotation of appositions, we loosely followed the instructions from MUC-7 (1998) and Davies et al. (1998). The MUC manual proposes to tag the NP as a whole as well as any separate NP contained in the appositive clauses, if the appositive clause is contiguous to the NP. We did not follow this proposal and tagged both NPs of the apposition as separate NPs, as for example in (22).

  (22) Volgens de krant De Morgen zijn er drie buitenlandse kandidaat-overnemers voor **ABX**, **het NMBS-filiaal dat het goederenvervoer over de weg verzorgt**.

  English: According to the newspaper De Morgen, there are three foreign candidates for taking over **ABX**, **the NMBS subsidiary that organizes road haulage**.

Furthermore, in contrast to the MUC guidelines, which stipulate that appositional phrases are not marked when they are negative or when there is only partial overlap of sets, we decided that also negative information is information which could be useful in for example an information extraction task and we therefore also marked these appositional phrases as in the example (23).

(23) **Karel Degucht**, **niet meteen een toonbeeld van bescheidenheid**, (...)

English: **Karel Degucht**, **not exactly a model of modesty**, (...)

Table 2.1 gives an overview of the proportion of each of the above described relation types for the complete KNACK-2002 corpus. Out of a total of 12,546 coreferentially annotated NPs, 9,277 refer to another noun phrase. Among these 9,277 coreferential links, the largest part, namely 98.5%, represent an identity relation. The other links only make up a small fraction of the total number of coreferences. However, this small number is also a result of the fact that only one antecedent is annotated per anaphor, which is illustrated in example (24).

(24) **Chirac** was in die tijd **de voorzitter van de RPR en burgemeester van Parijs**. (...) Medewerkers van **de president** beweren dat de terugkeer van Schuller een politiek manoeuvre van links is om **Chirac** onderuit te halen.

English: At that time **Chirac** was **president of the RPR and mayor of Paris**. (...) Staff members of **the president** claim that the return of Schuller is a political manoeuvre from the left to destabilize **Chirac**.

In the current annotation, *"de voorzitter van de RPR en burgemeester van Parijs"* is linked to *"Chirac"* and *"de president"* is also linked to *"Chirac"* and all these NPs are part of one single coreference chain. However, the transitivity assumption which is the base for each coreferential chain is violated here since *"voorzitter van de RPR en burgemeester van Parijs"* and *"de president"* do not refer to each other. One possible strategy to undo this violation is to annotate the type of relation between the anaphor and each of its antecedents, instead of only one antecedent. Another possible strategy is to leave the idea of transitivity and to explore the idea of file-keeping as proposed by Heim (1982).

Kibble (2000) and van Deemter and Kibble (2000) have criticized that the MUC coreferential annotations (MUC-7 1998) go well beyond the annotation of the relation of coreference, since both indefinite noun phrases and bound anaphors are coreferentially annotated. As in Davies et al. (1998) and Mitkov, Evans, Orasan, Barbu, Jones and Sotirova (2000), we mainly followed the MUC guidelines in the annotation of the Dutch texts, and the same criticism may be leveled against these annotations. However, by defining different new types of coreferential relations next to the "IDENT" relations, we took into account the critical

Table 2.1: Number of links per relation type.

Relation type	
IDENT	9,139
IDENT + TIME attribute	30
MOD	55
BOUND	43
ISA	40
Total	9,277

remarks of Kibble (2000) and van Deemter and Kibble (2000). By defining a "BOUND" type, for example, all bound anaphors can easily be traced.

2.3.3 Resulting data sets

Since it is the first time this corpus is experimented with, we decided not to use the whole corpus of 267 documents for the coreference resolution experiments. Instead, we made a random, but balanced selection of 50 documents covering different topics. We selected 10 documents covering internal politics, 10 documents on foreign affairs, another 10 documents on economy, 5 documents on health and health care, 5 texts covering scientific topics and finally 10 documents covering a variety of topics (such as sports, education, history and ecology). In total, the documents contain 25,994 words and 3,014 coreferential tags. Half of the texts was used as training set and the other half as test set. The division between testing and training material was done randomly at document level (in order to avoid documents being divided in two). The KNACK-2002 training and test set contain 1,688 and 1,326 anaphoric NPs, respectively.

2.4 Inter-annotator agreement

Since the annotation of coreferential relations is complex, it can lead to disagreement among the annotators. The degree of this disagreement largely depends on the scope of annotation. Poesio and Vieira (1998) show that narrowing the scope of annotation leads to a larger **inter-annotator agreement**. Mitkov et al. (2000) report on another project about anaphora annotation aiming at a large inter-annotator agreement at the expense of wide-coverage annotation. In order to reduce the number of annotation errors, many annotation schemes aim at reducing the complexity of the relations to be annotated. In MUC-6

(MUC-6 1995) and MUC-7 (MUC-7 1998), for example, coreferential relations are only marked between noun phrases, including definite and indefinite noun phrases, different types of pronouns and proper names. For MUC-6, the human inter-annotator agreement was in the low 80's, whereas the best coreference resolution systems obtained recall scores in the low 60's and precision scores in the low 70's (Hirschman, Robinson, Burger and Vilain 1997). In a small experiment in which 5 texts were annotated by two annotators, they show that only 16% of the errors are so-called hard errors for which there is no agreement about the antecedent for a referring expression. Based on these observations, some minor revisions were proposed for the MUC-7 annotation scheme. Although one of the criteria in the development of the MUC-7 annotation guidelines was the ability to achieve a good (95%) inter-annotator agreement, we were not able to find any numbers about the actual agreement.

For the annotation of the Dutch news magazine texts, the following strategy was taken. All texts were annotated by two annotators from a pool of five native speakers with a background in linguistics. Although we were interested in how quick these annotations could be produced, we were not able to determine the speed of the annotation process since this depends on factors such as the different annotation behaviour of the annotators, the familiarity with the annotation manual and the annotation tool and the different lengths of the texts. After the individual coreference annotation by both annotators, we decided not to work with possibly different annotations. This decision was based on the observations of Hirschman et al. (1997) that more than half (56%) of the errors were missing annotations and that 28% errors represented "easy" errors (such as the failure to mark headlines or predicating expressions). Instead, the annotators verified all annotations together in order to reach one single consensus annotation. In case of no agreement, the relation was not marked.

2.5 Summary

In this first chapter we introduced the coreferentially annotated data sets which will be used in our experiments: the English MUC-6 and MUC-7 data sets and the newly developed Dutch KNACK-2002 corpus. Having also discussed in this and the previous chapter the overlap and the differences between coreferential and anaphoric relations, we will use the terms interchangeably in the remainder of this thesis, as is also done in most of the work on computational coreference resolution. This choice can be motivated by the fact that our system is based on annotated corpora which do not clearly distinguish between coreferential and anaphoric relations (MUC-6 and MUC-7). Furthermore, we showed that the non-identity relations only represent a small fraction of the annotations in the

KNACK-2002 corpus, which further justifies our choice.

In the following chapter, we will continue with a description of the relevant information sources for the resolution of coreferential relations.

CHAPTER 3

Information sources

In supervised learning of coreference resolution, one is given a training set containing labeled instances. These instances consist of attribute/value pairs which contain possibly disambiguating information for the classifier, whose task it is to accurately predict the class of novel instances. A good set of features is crucial for the success of the resolution system. An ideal feature vector consists of features which are all highly informative and which can lead the classifier to optimal performance. This implies that irrelevant features should be avoided, since the learner can have difficulty in distinguishing them from the relevant features when making predictions. Furthermore, it is important to keep the attribute noise as low as possible, since errors in the feature vector can heavily affect the predictions.

This chapter deals with the problem of the selection of information sources for the resolution of coreferential relations. The first section (3.1.1) discusses the preparation of the data sets. We describe the different preprocessing steps that were taken for the construction of the training and test corpora. In Section 3.1.2, we briefly mention the problem of the selection of positive and negative instances and the related problem of the skewed class distributions (Chapter 7 will extensively deal with the problem of highly skewed training data). In Section 3.1.3, we explore the use of three different data sets, viz. one for the pronouns, one

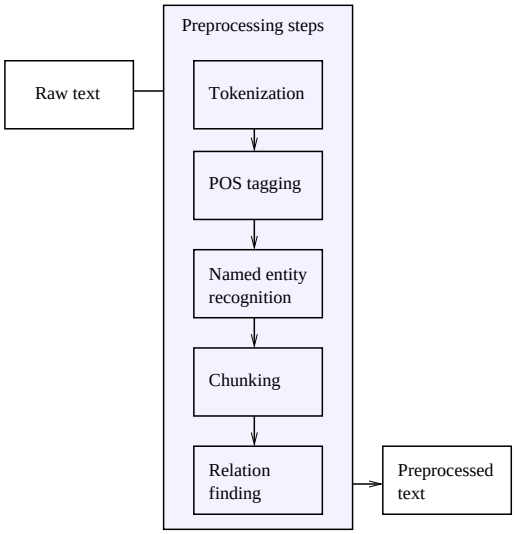
for the named entities and a third data set for the other noun phrases, instead of one single data set. Section 3.2 gives an overview of the information sources which have been used in other work on coreference resolution. In this overview, we focus on the shallow information sources which can easily be computed. We continue with a description of the different features which we used for our experiments.

3.1 Data preparation

3.1.1 Preprocessing

For the construction of the initial data sets, we selected all noun phrases in the MUC-6/7 and KNACK-2002 corpora. These noun phrases could be detected after preprocessing the raw text corpora. Figure 3.1 schematically displays the preprocessing procedure for the test and training corpora.

Figure 3.1: Preprocessing procedure for the training and test corpora.



The following preprocessing steps were taken for both English and Dutch:

- **Tokenization** means that punctuation is split from adjoining words. Verb contractions (e.g. won't) and the genitive of nouns (children's) are split

into their component morphemes, and each morpheme is tagged separately. The tokenization for both Dutch and English was performed by a rule-based system using regular expressions.

- **Named entity recognition** involves the detection of names in text. For the recognition of these names in the English data sets, a memory-based learning[1] approach (Demeulder and Daelemans 2003) was used. This system distinguishes between three types of named entities, viz. persons, organizations and locations. Dutch named entity recognition, on the other hand, was performed by looking up the entities in lists of location names, person names, organization names and other miscellaneous named entities.

- **Part-of-speech tagging and text chunking** for English was performed by the memory-based tagger MBT (Daelemans, Zavrel, Berck and Gillis 1996, Daelemans, Zavrel, van den Bosch and van der Sloot 2003), which was trained on text from the Wall Street Journal corpus in the Penn Treebank (Marcus et al. 1993), the Brown corpus (Kucera and Francis 1967) and the Air Travel Information System (ATIS) corpus (Hemphill, Godfrey and Doddington 1990). During text chunking syntactically related words were combined into non-overlapping phrases. Although the chunker provided different types of phrases, we were mainly interested in the NP chunks. These NP chunks are base NPs which contain a head, optionally preceded by premodifiers, such as determiners and adjectives. Postmodifiers are not part of the noun phrase. This implies that in example sentence (25), "Union representatives" is selected as NP, instead of "Union representatives who could be reached", which is considered as one single NP in the MUC annotation.

  Part-of-speech tagging and text chunking for Dutch was again performed by the memory-based tagger MBT, this time trained on the Spoken Dutch Corpus (CGN)[2], a 10-million word corpus of spoken Dutch. The part-of-speech classes of the CGN corpus are rich. Apart from defining that a word is a pronoun (VNW), a verb (WW) or something else, a part-of-speech tag contains several other features of the word, as illustrated in the following sentence from the KNACK-2002 corpus.

  > Woensdag/N(eigen,ev,basis,zijd,stan) waren/WW(pv,verl,mv)
  > gevechten/N(soort,mv,basis) uitgebroken/WW(vd,vrij,zonder)
  > tussen/VZ(init) aanhangers/N(soort,mv,basis) van/VZ(init)
  > twee/TW(hoofd,prenom,stan) lokale/ADJ(prenom,basis,met-e,stan)

[1] An elaborate description of memory-based learning is provided in Section 4.2

[2] More information on this corpus can be found at http://lands.let.ru.nl/cgn/. All words in the corpus are annotated with part-of-speech information and about 10% of the sentences are annotated with syntactic trees.

rivalen/N(soort,mv,basis) ./.

English: On Wednesday, there were fights between followers of two local rivals.

In the following KNACK-2002 training set sentence, an example is given of the Dutch NP chunking. For the construction of the feature vectors, we are mainly interested in the NP chunk, since our focus is on the resolution of coreferential relations between noun phrases.

Vanaf [de jaren tachtig] werd [de situatie] nog gevaarlijker : [India] en [Pakistan] beschikten over [atoomwapens] .

English: Since the beginning of the eighties, the situation became even more dangerous: both India and Pakistan had nuclear weapons.

- **Grammatical relation finding**. The relation finder of the shallow parser determines which chunk has which grammatical relation to which verbal chunk, e.g. subject, object, etc.

  The relation finder for English (Buchholz, Veenstra and Daelemans 1999, Buchholz 2002) is trained on sections 10 to 19 of the WSJ Corpus of the Penn Treebank II. These sections contain 515,390 tokens and 21,747 sentences. It has 381 output classes, among which we are primarily interested in the subject and object classes. The output of the relation finder is restricted to one relation per word.

  For Dutch, the relation finder (Tjong Kim Sang, Daelemans and Höthker 2004) is trained on the previously mentioned Spoken Dutch Corpus (and more specifically on 2,381,405 words or 153,937 sentences). It offers a fine-grained set of grammatical relations, such as modifiers, verbal complements, heads, direct objects, subjects, predicative complements, indirect objects, reflexive objects, etc. The output tags of the relation finder have the format "I-XXX*I-YYY" where I-XXX represents the relation of the word to the first verb in the sentence and I-YYY represents its relation to the second verb. Also for Dutch, we are mainly interested in the subjects, objects and predicative constituents for the construction of features for our task of coreference resolution.

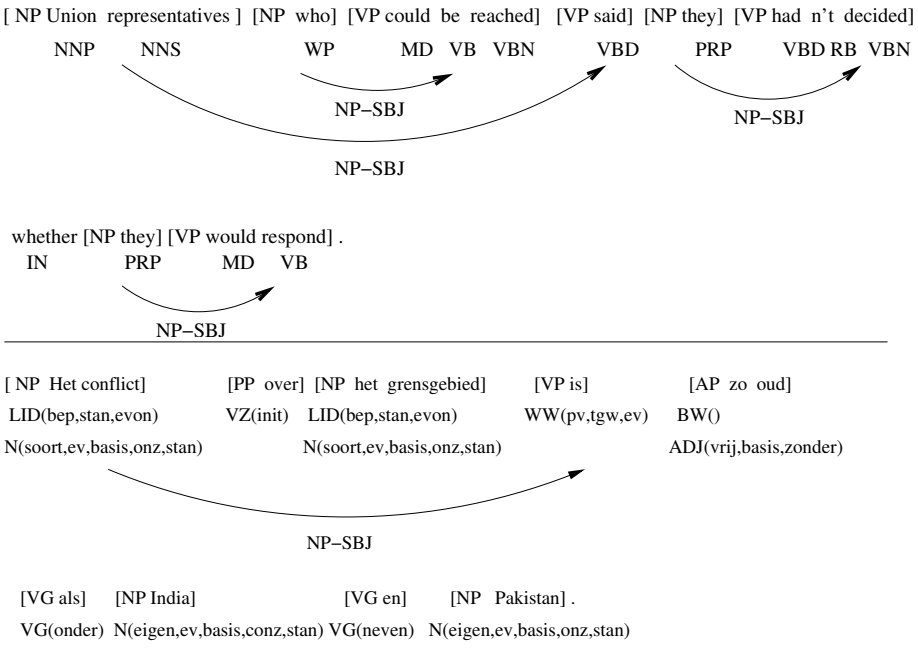
  Figure 3.2 gives an overview of the part-of-speech tags, chunk tags and relation tags for MUC-6 training sentence (25) and the KNACK-2002 training sentence (26).

(25) < COREF ID =”11” MIN = ”representatives” > < COREF ID = ”8” TYPE = ”IDENT” REF = ”9” > Union < /COREF > representatives who could be reached < /COREF > said < COREF ID = ”10” TYPE = ”IDENT” REF = ”11” > they < /COREF > hadn’t decided whether < COREF ID = ”12” TYPE = ”IDENT” REF = ”10” > they < /COREF > would respond.

(26) < COREF ID = ”1528” MIN = ”conflict” > Het conflict over het grensgebied < /COREF > is zo oud als < COREF ID = ”1464” > < COREF ID = ”1451” > India < /COREF > en < COREF ID = ”1459” > Pakistan < /COREF > < /COREF>.

English: The conflict about the border area is as old as India and Pakistan.

Figure 3.2: Part-of-speech tags, chunk tags and relation tags for example sentences (25) and (26).



- **Morphological analysis (only for Dutch)** For Dutch we also performed a machine learned morphological analysis (De Pauw, Laureys, Daelemans and Van hamme 2004). Dutch features a more extensive inflection, conjugation and derivation system than English. An example is the use of diminutive suffixes, such as in the following KNACK-2002 sentence:

  (27) Dus vroeg < COREF ID = "65" TYPE = "IDENT" REF = "64" > hij < /COREF > het aan de directeur van < COREF ID = "61" TYPE = "IDENT" REF = "55" > ISI < /COREF > , < COREF ID = "322" MIN = "bureautje" TYPE = "IDENT" REF = "61" > toen een **bureautje** dat over militaire attachés van buitenlandse ambassades ging en over zaken met < COREF ID = "62" TYPE = "IDENT" REF = "52" > India < /COREF > < /COREF > , met een director general en zes of zeven officieren . < COREF ID = "323" TYPE = "IDENT" REF = "322" MIN = "kantoor" > **Dat kleine kantoor** < /COREF > werd een groot operatiecentrum.'

  English: So he asked the manager of ISI, in those days **a small office** that was in charge of military attachés of foreign embassies and of affairs with India, with one general director and six or seven officers. **That small office** became a large operating center.

  Since diminutives are not listed in the EuroWordNet database (described in Section 3.2) which we will use for the construction of our semantic synonym and hypernym features, we needed a procedure to strip off the "tje" from "bureautje". Furthermore, also compounds had to be split into their components. Compounding in Dutch can occur through concatenation as in "pensioenspaarfonds" (English: "pension saving fund") and through concatenation in combination with the infix /s/ as in "bedrijfsstructuur" (English: "company structure") or in combination with the /e<n>/ infix as in "studentenorganisatie" (English: "student organization") and "studentenkoepel" (English: "student umbrella organization").

The information obtained through this preprocessing will be used in the construction of the feature vectors for our learning techniques. We will now describe how these feature vectors are built.

3.1.2 Positive and negative instances

On the basis of the preprocessed texts, we selected positive and negative instances for the training data and test instances for the test data. For the con-

Table 3.1: Number of positive and negative instances for each data set.

Instances	MUC-6	MUC-7	KNACK-2002
Train positive	11,266	8,815	6,457
Train negative	159,815	143,070	95,919

struction of the test data, we refer to Chapter 8, which is completely devoted to the testing procedure. For the cross-validation experiments, both learning methods require instances from at least two classes[3]. The annotated coreferential links serve as basis for the construction of the first class, the positive instances. These **positive instances** were made by combining each anaphor with each preceding element in the coreference chain. The **negative instances** were built as follows. For the MUC data sets they were built (i) by combining each anaphor with each preceding NP which was not part of any coreference chain and (ii) by combining each anaphor with each preceding NP which was part of another coreference chain. Since the KNACK-2002 data contain some large documents, exceeding 100 sentences, this negative instance construction gave us a negative instance base of more than 300,000 instances, opposed to the 7,863 positive instances. Therefore, we only made negative instances for the NPs in a range of 20 sentences preceding the candidate anaphor. The number of instances after application of these criteria is displayed in Table 3.1. In case of the MUC-6 training data set, for example, with its 1,644 references a training instance base was built consisting of 171,081 instances. And a training instance base of 102,376 instances was built for the 1,687 references in the KNACK-2002 training data. An example of the construction of the training instances is given in Table 3.2.

Table 3.1 and Table 3.2 already reveal **the highly skewed class distribution**, which is caused by a small number of positive instances and a large number of negative instances. In MUC-6, for example, 159,815 instances out of 171,081 are negative and merely 11,266 (6.6% of the total) are positives. In MUC-7, only 8,815 out of 151,885 (5.8% of the total) are positive. And in KNACK-2002, merely 6.3% of the instances is classified as positive. Furthermore, the number of instances in the data sets is large compared to the number of references (in MUC-6 1,644, MUC-7 1,905 and KNACK-2002 1,687) present in both sets. In Chapter 7, we will discuss the different strategies used by Ng and Cardie (2002a), Soon et al. (2001), Yang et al. (2003) and others to alleviate this imbalance. We will also investigate whether the highly skewed class distribution hinders classification accuracy for our learning techniques and we will investigate different techniques to tackle the problem of the highly skewed class distribution.

[3]For a discussion on one-class classification we refer to Chapter 7

Table 3.2: NP pairs for which instances are built for the sentences "Eastern Airlines executives notified union leaders that the carrier wishes to discuss **selective wage reductions** on **Feb. 3**. **Union representatives who could be reached** said **they** hadn't decided whether **they** would respond." The NPs in front of the arrow represent the candidate anaphors; the NPs behind the arrow represent the candidate antecedents. The text is processed from right to left.

they => they	POS
they => Union representatives who could be reached	POS
they => Union	NEG
they => Feb. 3	NEG
they => selective wage reductions	NEG
they => wage	NEG
they => union	NEG
they => Eastern Airlines	NEG
they => Union representatives who could be reached	POS
they => Union	NEG
they => Feb. 3	NEG
they => selective wage reductions	NEG
they => wage	NEG
they => union	NEG
they => Eastern Airlines	NEG
Union representatives who could be reached => Union	NEG
Union representatives who could be reached => Feb. 3	NEG
Union representatives who could be reached => selective wage reductions	NEG
Union representatives who could be reached => wage	NEG
Union representatives who could be reached => union	NEG
Union representatives who could be reached => Eastern Airlines	NEG
Union => Feb. 3	NEG
Union => selective wage reductions	NEG
Union => wage	NEG
Union => union	POS
Union => Eastern Airlines	NEG
Feb. 3 => selective wage reductions	NEG
Feb. 3 => wage	NEG
Feb. 3 => union	NEG
Feb. 3 => Eastern Airlines	NEG
selective wage reductions => union	NEG
selective wage reductions => Eastern Airlines	NEG
wage => Eastern Airlines	NEG

3.1.3 One vs. three

Instead of merging the different types of NPs into one single training and test set (as for example Ng and Cardie (2002a) and Soon et al. (2001)), we built 3 smaller datasets. By analogy with the so-called word-expert approach which proved to be successful in word sense disambiguation (Veenstra, van den Bosch, Buchholz, Daelemans and Zavrel 2000, Hoste, Hendrickx, Daelemans and van den Bosch 2002), in which a specialized system is developed for each single ambiguous word, we built three systems, each specialized in one NP type. This resulted in a learning system for pronouns, one for named entities and a third system for the other NPs.

The main motivation for this approach is that other information sources play a role in the resolution of pronominal references than for example in the resolution of references involving proper nouns. Example sentence (28) clearly shows the importance of string matching or aliasing in the resolution of proper nouns. These features are less important for the resolution of the first coreferential link between a pronoun and a common noun NP in example (29), for which information on gender, number and distance is crucial.

(28) **Eastern Air** Proposes Date For Talks on Pay-Cut Plan. **Eastern Airlines** executives notified union leaders (...) By proposing a meeting date, **Eastern** moved one step closer toward reopening current high-cost contract agreements with its unions.

(29) **Union representatives who could be reached** said **they** hadn't decided whether **they** would respond.

In order to test our hypothesis of three classifiers, each trained on one specific NP type, being better than one single classifier, we built the data sets displayed in Table 3.3. The 'Pronouns' data set contains the NPs ending on a personal, reflexive or possessive pronoun. The 'Proper nouns' data set contains the NPs which have a proper noun as head, whereas the 'Common nouns' data set contains all other NPs which are not in the two other categories. And the fourth dataset is the sum of all three datasets.

Additional motivation for the construction of three different data sets was found in the results reported by Harabagiu et al. (2001), Ng and Cardie (2002a) and Strube et al. (2002). Ng and Cardie (2002a) calculated the performance of their system on pronouns, proper nouns and common nouns and observed a low precision on common noun resolution (antecedents were searched for many non-anaphoric common nouns) and a high precision on pronoun and proper noun resolution. A similar conclusion was made by Strube et al. (2002) when

Table 3.3: Number of instances per NP type in the MUC-6/7 and KNACK-2002 training corpora.

	MUC-6	
NP type	train positive	train negative
Pronouns	2,006	26,811
Proper nouns	5,901	68,634
Common nouns	3,359	64,370
Complete	11,266	159,815

	MUC-7	
NP type	train positive	train negative
Pronouns	2,705	28,952
Proper nouns	3,455	54,109
Common nouns	2,655	60,009
Complete	8,815	143,070

	KNACK-2002	
NP type	train positive	train negative
Pronouns	3,111	33,155
Proper nouns	2,065	31,370
Common nouns	1,281	31,394
Complete	6,457	95,919

experimenting with C5.0 (Quinlan 1993) on a corpus of German texts. These experiments show that the feature describing the form of the anaphor (definite NP, indefinite NP, personal pronoun, demonstrative pronoun, possessive pronoun, proper noun) is the most important. They furthermore show that the classifier performs poorly on definite NPs and demonstrative pronouns, moderately on proper nouns and quite good on personal pronouns and possessive pronouns. These different error rates reported for the different types of NPs are an additional motivation for building more fine-grained data sets for each NP type.

This grouping of the different types of NPs does not only allow for building more specialized classifiers and making error analysis more transparent (see Chapter 8). It also makes comparison between our two studied languages, English and Dutch, easier. For example, one of the major problems for both languages in case of pronoun resolution is that some pronouns do not always refer to a referent. These pronouns are called "pleonastic pronouns", such as the English "it" and

the Dutch "het". As will be shown in Chapter 8, these pleonastic pronouns are the main source of errors in the English pronoun resolution system. For Dutch, pronominal resolution is even more complex: Dutch third person female and male personal and possessive pronouns can not only refer to living creatures, but also to objects, organizations, as in (30, 31).

(30) **India** van **zijn** kant oefende en oefent een harde repressie uit tegen al wie ook maar enigszins verdacht was.

English: **India**, for **its** part, (...)

(31) **De ISI**, de Pakistaanse militaire geheime dienst, had de oorlog en de Afghanistanpolitiek volledig in handen. **Haar** chef, generaal Hamid Gul, leidde de operaties in eigen persoon.

English: **The ISI**, the Pakistan secret service, had complete control over the war and the Afghanistan politics. **Its** chief, (...)

Furthermore, the third person female personal pronoun and the third person plural personal pronoun both can take the same form in case of nominative use: "ze" or ""zij" (32, 33). A similar tendency can be observed in (34) for the demonstrative pronoun "die".

(32) **De langdurige perioden van militair bewind** hadden tot nefast gevolg dat **ze** het ontstaan van een cultuur van politieke democratie beletten.

English: **The long periods of military rule** have the pernicious consequence, that they inhibit (...)

(33) Toen verkiezingen **de dochter van Zia's slachtoffer, Benazir Bhutto**, aan de macht brachten, was **zij** niet in staat de generaals een nieuwe Afghanistanpolitiek op te leggen.

English: When elections brought to power **Benazir Bhutto, the daughter of Zia's victim**, **she** failed to impose (...)

(34) Pakistan bevroor de tegoeden van **twee moslimgroepen**. **Die** staan allebei op de Amerikaanse zwarte lijst wegens banden met al-Qaeda, het netwerk van **Osama Bin Laden**. En **die** kan bovendien ook in Kashmir zijn ondergedoken.

> English: Pakistan froze the assets of two Muslim groups. These are both listed on the American black-list because of ties with al-Quaeda, the organization of **Osama Bin Laden**. And **he** can also (...)

We will return to this issue in the error analysis in Chapter 8.

3.2 Selection of informative features

Several information sources contribute to a correct resolution of coreferential relations, viz. morphological, lexical, syntactic, semantic and positional information and also world-knowledge. In this section, we give an overview of the information sources which have been used in other machine learning work on coreference resolution. In this overview, we will focus on the machine learning approaches using shallow information sources which can be easily computed. We continue with a description of the different features used for our experiments.

3.2.1 The choice of features in related work

Let us first have a look at some characteristics of the feature vectors in previous machine learning work on coreference resolution.

- All systems use a **combination of lexical, syntactic, semantic and positional features**. There are however large differences in the number of features which are used in the different systems. The anaphora resolution system for Japanese of Aone and Bennett (1995), one of the first machine learning approaches to anaphora resolution, for example, uses 66 features, whereas the RESOLVE system of McCarthy and Lehnert (1995) makes its predictions based on 8 features.

- Furthermore, all systems distinguish between **'unary' features**, describing characteristics from a single anaphor or from its (candidate) antecedent and **'binary' features**, describing the characteristics of the relation between an anaphor and its (candidate) antecedent. Another type of binary features has been proposed by Yang et al. (2004a), who incorporate in the feature vector a set of features describing the antecedent of the candidate antecedent.

- Whereas in the early work (e.g. Aone and Bennett (1995)) **the usefulness of the different features** has not been evaluated, we can see a tendency

in recent work (such as Soon et al. (2001), Ng and Cardie (2002c) and Yang, Su, Zhou and Tan (2004b)) to assess the informativeness of the features. Soon et al. (2001), for example, study the contribution of the features by training their system only taking into account one single feature and some combinations of features. And Strube and Müller (2003) use an iterative feature selection procedure in a spoken dialogue corpus.

We will now continue with a description of the features used in the resolution systems of McCarthy and Lehnert (1995), McCarthy (1996), Fisher et al. (1995), Cardie and Wagstaff (1999), Soon et al. (2001), Ng and Cardie (2002c), Strube et al. (2002) and Yang (2003,2004,2004b). We start with a short description of the features used in these systems and then give a schematic overview of the different types of features.

The **RESOLVE** system from McCarthy and Lehnert (1995), McCarthy (1996) and Fisher et al. (1995) was one of the first machine learning approaches to coreference resolution. The first RESOLVE system (McCarthy and Lehnert 1995) was used for the resolution of coreferences in the MUC-5 English Joint Venture corpus. For this MUC-5 experiment, 8 features were used, among which 2 features focusing on the topic of joint ventures (a. does reference i refer to a joint venture child, a company formed as a result of two or more entities?, b. do both references refer to a joint venture child?). A new version of the RESOLVE system (McCarthy 1996, Fisher et al. 1995) was also applied to the MUC-6 coreference task. This version used 39 features, among which features based on proper name recognition, syntactic analysis, string matching and noun phrase analysis.

The unsupervised approach from Cardie and Wagstaff (1999) views coreference resolution as a clustering task. This unsupervised learning system takes as input the individual words, the head noun, the position of the NP, 3 features describing the type of NP, number and gender information and semantic information. It was also applied on the MUC-6 data.

The system of **Soon et al. (2001)** uses the C4.5 decision tree learner and takes as input vectors consisting of 12 features, all of which already have been used by McCarthy (1996), Fisher et al. (1995) or Cardie and Wagstaff (1999). The detection of all possible antecedents and anaphors in the input text is done after application of several natural language processing modules, viz. tokenization, sentence segmentation, morphological processing, part-of-speech tagging, noun phrase identification, named entity recognition, nested noun phrase extraction and semantic class determination. The feature vectors consist of relatively shallow information sources among which 5 features indicate the type of noun phrase

(pronoun, definite noun phrase, demonstrative noun phrase, proper noun). The other features provide information on distance, gender and number agreement, semantic class agreement, string matching, aliasing and appositions. Soon et al. (2001) also study the contribution of each single feature by training their system on this feature and they show that even with a limited set of three features, their decision tree learning system obtains results near to the highest scores already reported on these two data sets.

Ng and Cardie (2002c) explore the effect of including additional lexical, semantic and grammatical potentially useful knowledge sources on top of the features used by Soon et al. (2001) for their coreference resolution classifier. These features were not derived empirically from the corpus, but were based on common-sense knowledge and linguistic intuitions regarding coreference. They trained C4.5 and RIPPER on MUC-6 and MUC-7. They show that the expansion of the feature set leads to a decrease in precision, especially for the common nouns, mainly caused by the application of low-precision rules. They suggest that data fragmentation has contributed to the drop in performance. In order to improve precision scores, they perform manual feature selection, discarding features used primarily to induce low-precision rules for common noun resolution and they retrain the classifier using the reduced feature set. This selection leads to a restricted feature set of 18 additional features to those proposed by Soon et al. (2001).

Strube et al. (2002) report results with C5.0 (a version of C4.5) on a corpus consisting of short German texts. They introduce a new feature for both the anaphor and the possible antecedent: minimum edit distance (MED) (Levenhstein 1966, Wagner and Fisher 1974). The minimum edit distance between two NPs is defined as the minimum number of deletions, insertions, and substitutions required to transform one NP into the other. They show that the use of this feature leads to a significant performance improvement for the definite NPs and proper names.

Yang (2004,2004b) report results with C5.0 on the use of two additional types of features. Yang et al. (2004a) use additional features in the feature vector which describe the antecedent of the candidate. And Yang et al. (2004b) explore the usefulness of adding additional cluster features to the feature vector of a given anaphor. These cluster features describe the relationship (such as number and gender agreement, string similarity, etc.) between the candidate anaphor and a cluster of possibly referring NPs. In an analysis of the decision trees produced with and without cluster information, they show that string matching is crucial in both systems. We will return to this issue of feature importance in the next Chapter.

All but one of the approaches described above rely on annotated data; only the clustering approach from Cardie and Wagstaff (1999) works with unlabeled data. Since manual labeling of data is very time- and labour-intensive, NLP research has recently started to focus on cheap methods for augmenting the available training data. Ng and Cardie (2003) and Müller, Rapp and Strube (2002) report on this type of research applied to the specific problem of coreference resolution. Both Müller et al. (2002) and Ng and Cardie (2003) apply co-training (Blum and Mitchell 1998) to coreference resolution. In co-training, a large unlabeled sample is used to boost performance of a learning algorithm when only a small set of labeled examples is available. The co-training algorithm consists of two classifiers, both trained on a different feature subset (a so-called "view") of the training data. The co-training algorithm is supposed to bootstrap by gradually extending the training data with self-labeled instances. Although co-training has been successfully used in document classification (Blum and Mitchell 1998) and different other domains, Müller et al. (2002) and Ng and Cardie (2003) report no or little accuracy improvements when using co-training on their coreference data. As a possible explanation for these results, they mention the difficulty to split the available features into two compatible and uncorrelated feature sets. As a possible solution Ng and Cardie (2003) propose a single-view bootstrapping method in which the algorithm uses two difference learning algorithms to train two classifiers on the same feature set. The highest scored instances produced by one classifier are then added to the training set of the other classifier and vice versa. They report an initial rise in F-measure followed by a gradual deterioration (caused by pollution of the labeled data).

We will now continue with a schematic overview of the features used in the systems mentioned above. For clarity reasons, we divided the features into different categories: morphological and lexical features, positional features, syntactic features, string-matching features and semantic features. We will not take into account minimal variations to the listed features nor "special-purpose features" which were designed for a specific data set, such as the features incorporating information on joint ventures for the MUC-5 English Joint Venture corpus.

Positional features. The positional features give information on the location of the candidate anaphors, the candidate antecedents and also inform on the distance between both noun phrases. In the literature, both binary (yes/no) and numeric values (e.g. 0 if both constituents occur in the same sentence) have been used for these features.

Do the two NPs occur in the same sentence?	yes/no	(McCarthy and Lehnert 1995, McCarthy 1996, Fisher et al. 1995)
Do the two NPs occur in adjacent sentences?	yes/no	(McCarthy 1996, Fisher et al. 1995)
Position of the anaphoric NP counted in number of NPs starting at the beginning of the document	1...n	(Cardie and Wagstaff 1999)
Distance between the two NPs in terms of the number of sentences	0,1...(n or >1)	(Soon et al. 2001, Ng and Cardie 2002c, Strube et al. 2002, Yang et al. 2003)
Distance between the two NPs in terms of the number of paragraphs	0...n	(Ng and Cardie 2002c, Yang et al. 2003)
Distance between the two NPs in terms of the number of words	1...n	(Strube et al. 2002)
Distance between the two NPs in terms of the number of markables	1...n	(Strube et al. 2002)
Is the NP a title?	yes/no	(Ng and Cardie 2002c, Yang et al. 2003)
(...)		

Morphological and lexical features. It is the task of morphology to describe the internal grammatical structure of words. An example of the morphological feature is the feature checking for the number of a certain word. For the construction of the lexical features, you search the NPs for certain keywords, e.g. “the” as first word of the NP, ...

Does the NP start with an definite article?	yes/no	(McCarthy 1996, Fisher et al. 1995, Soon et al. 2001, Ng and Cardie 2002c, Yang et al. 2003)
Does the NP start with an indefinite article?	yes/no	(McCarthy 1996, Fisher et al. 1995, Yang et al. 2003)

Is the NP definite, indefinite or none of both?	indef/ def/ none	(Cardie and Wagstaff 1999)
Does the NP start with a demonstrative pronoun?	yes/no	(Soon et al. 2001, Ng and Cardie 2002c)
What is the form of the NP?	def/indef/pers. pron./dem. pron./poss. pron./name	(Strube et al. 2002)
Is the NP pronominal?	yes/no	(McCarthy 1996, Fisher et al. 1995, Soon et al. 2001, Ng and Cardie 2002c, Yang et al. 2003)
Are both NPs pronominal?	yes/no	(Ng and Cardie 2002c)
Is the anaphoric NP a pronoun and is the candidate antecedent NP its antecedent according to a naive pronoun resolution algorithm?	yes/no	(Ng and Cardie 2002c)
What is the number of the head noun of the NP?	plural/sing	(Cardie and Wagstaff 1999)
Do the NPs agree in number?	yes/no	(Soon et al. 2001, Ng and Cardie 2002c, Yang et al. 2003)
What is the gender of the NP?	masc/fem/ either/ neuter	(Cardie and Wagstaff 1999)
Do the NPs agree in gender?	yes/no/ unknown	(Soon et al. 2001, Ng and Cardie 2002c, Yang et al. 2003)
Do the NPs agree in both gender and number?	yes/no/ unknown	(Ng and Cardie 2002c, Strube et al. 2002)
(...)		

Syntactic features. The syntactic features inform on the function (e.g. subject, object, appositive, etc.) of the anaphoric or antecedent noun phrase in the sentence.

Are the NPs in different complement roles of the same verb	yes/no	(McCarthy 1996, Fisher et al. 1995)

Is the NP a subject	yes/no	(Ng and Cardie 2002c)
Are both NPs subjects?	yes/no	(McCarthy 1996, Fisher et al. 1995)
Is the antecedent NP the most recent compatible (in gender and number) subject?	yes/no	(Fisher et al. 1995)
Are both NPs in the same constituent?	yes/no	(McCarthy 1996, Fisher et al. 1995)
Do the phrases share a common head noun?	yes/no	(McCarthy 1996, Fisher et al. 1995)
Do the phrases share a common modifier?	yes/no	(McCarthy 1996, Fisher et al. 1995)
Do the NPs share a common head noun or a common modifier?	yes/no	(McCarthy 1996, Fisher et al. 1995)
Do the NPs share a common, simple NP?	yes/no	(McCarthy and Lehnert 1995, McCarthy 1996, Fisher et al. 1995)
Is the pronoun nominative, accusative, possessive or ambiguous or is the NP non pronominal?	nom/ acc/ poss/ amb/ none	(Cardie and Wagstaff 1999)
Is the NP an appositive?	yes/no	(Cardie and Wagstaff 1999, Soon et al. 2001, Ng and Cardie 2002c, Yang et al. 2003)
Is the NP indefinite and not appositive?	yes/no	(Ng and Cardie 2002c)
Do the NPs form a predicate nominal construction?	yes/no	(Ng and Cardie 2002c)
Do the NPs have the maximal NP projection?	yes/no	(Ng and Cardie 2002c)
Does one NP span the other?	yes/no	(Ng and Cardie 2002c)
Do the NPs violate conditions B or C of the Binding Theory?	yes/no	(Ng and Cardie 2002c)
Can the NPs be co-indexed based on simple heuristics?	yes/no	(Ng and Cardie 2002c)
Do the NPs have compatible values for the four preceding conditions?	yes/no	(Ng and Cardie 2002c)
Is the antecedent an embedded noun?	yes/no	(Ng and Cardie 2002c, Yang et al. 2003)

What is the grammatical function of the NP?	subject/ object/ other	(Strube et al. 2002)
Do both constituents share the same grammatical function?	yes/no	(Strube et al. 2002)
(...)		

String-matching features The following features are all selected on the basis of string matching.

Are the phrases identical?	yes/no	(McCarthy 1996, Fisher et al. 1995, Strube et al. 2002)
Is the anaphoric NP a substring of the antecedent NP?	yes/no	(McCarthy 1996, Fisher et al. 1995, Strube et al. 2002)
Do the NPs match after stripping of articles and demonstrative pronouns?	yes/no	(Soon et al. 2001, Ng and Cardie 2002c)
Are both NPs pronominal and the same string?	yes/no	(Ng and Cardie 2002c)
Are both NPs proper names and the same string?	yes/no	(Ng and Cardie 2002c)
Are both NPs non-pronominal and does the string of the antecedent match that of the anaphor?	yes/no	(Ng and Cardie 2002c)
(...)		

Semantic features The semantic features include features especially designed for the proper names or named entities. These features will inform on the type of named entity: person, location, company, etc. For the construction of other semantic features, a semantic network such as WordNet (Fellbaum 1998) can be used. These features will inform on animacy or will indicate whether the given noun phrase is male or female, whether it denotes an organization or a date, etc.

Does the NP contain a name?	yes/no	(McCarthy and Lehnert 1995, McCarthy 1996, Fisher et al. 1995, Cardie and Wagstaff 1999)
Does each NP contain the same name?	yes/no	(McCarthy 1996, Fisher et al. 1995, Soon et al. 2001, Ng and Cardie 2002c)
Does each NP contain a different name?	yes/no	(McCarthy 1996, Fisher et al. 1995)
Is the anaphoric NP an alias of the antecedent NP?	yes/no	(McCarthy and Lehnert 1995, McCarthy 1996, Fisher et al. 1995, Soon et al. 2001, Ng and Cardie 2002c, Yang et al. 2003)
Does the NP contain location information?	yes/no	(McCarthy 1996, Fisher et al. 1995)
Do the phrases have compatible location information?	yes/no/ unknown	(McCarthy 1996, Fisher et al. 1995)
What is the semantic class of the NP?	time/ city/ animal/ human/ object	(Cardie and Wagstaff 1999)
What is the semantic class of the NP?	human / concrete object / abstract object	(Strube et al. 2002)
What is the semantic class of the NP?	female/ male/ person/ organization/ date/ time/ money/ percent/ object	(Soon et al. 2001, Ng and Cardie 2002c)
Is the NP animate or not?	anim/inanim	(Cardie and Wagstaff 1999)
Do the NPs match in animacy?	yes/no	(Ng and Cardie 2002c)
(...)		

In the overview above, we focused on the features used in the machine learning approaches to coreference resolution. There are, however, numerous other features which could be informative in the task of coreference resolution, such as the individual words as used in Cardie and Wagstaff (1999), topicalization,

information on synonymy and hyperonymy / hyponymy as used by Vieira and Poesio (2000), collocation pattern preference as in Dagan et al. (1995), etc. In the following section, we will describe the features we used to build the feature vectors for our coreference resolution system.

3.2.2 Our shallow information sources

In this section, we given a description of the features we selected for our coreference resolution system. Whereas most of the discussed features are "classical" features used by most of the other machine learning resolution systems, we will also introduce some new features, especially semantic features, such as the hypernym and synonym features.

Since it is our objective to build a completely automated coreference resolution system, we only use so-called shallow information sources, namely information sources which are easy to compute. For example, for the construction of the syntactic features which inform on the syntactic category of a given noun phrase, we use the output of the shallow parser (described in Section 3.1). We will now continue with a description of the different features. A schematic overview is presented in Table 5.6. If the features for the Dutch data set differ from the features for the English data sets, this will be mentioned. If no language specification is given, this implies that the data sets of both languages share the same type of feature.

POSITIONAL FEATURES give information on the location of the candidate antecedents. We made use of the following three positional features:

- DIST_SENT: This feature gives information on the number of sentences between the candidate anaphor and its candidate antecedent. The values of this feature range from 0 (same sentence) and the number of sentences in the text minus 1.
- DIST_NP: This feature gives information on the number of noun phrases between the candidate anaphor and its candidate antecedent. The values of this feature range between 1 (NP immediately preceding the anaphor) and the number of NPs in the text minus 1.
- DIST_LT_THREE (values: 'yes', 'no') is a positional feature which has not been used by others before. This feature can take the following two values: 'yes' if both constituents are less than three sentences apart from one another and 'no' if both constituents are more than three sentences apart. The main motivation for the use of this feature is that a large

majority of the pronominal anaphors refer to an antecedent within a scope of three sentences (as shown in Chapter 8). Therefore, we created a feature which checks whether the antecedent is located within a scope of three sentences or not. This binary feature is more coarse-grained than the numeric DIST_SENT feature.

LOCAL CONTEXT FEATURES

- 12 features give information on the three words preceding and following the candidate anaphor, with their corresponding part-of-speech tags.

MORPHOLOGICAL AND LEXICAL FEATURES

- I_PRON, J_PRON and I+J_PRON (all three have values 'yes' and 'no') indicate whether a given candidate anaphor, its candidate antecedent or both are pronouns (personal, possessive, demonstrative or reflexive).
- J_PRON_I_PROPER can take three values ('yes', 'no' and 'na') and indicates whether the possible antecedent of an anaphoric pronoun is a proper noun. 'Na', which stands for "not applicable" is used in case of a non-pronominal anaphor.
- J_DEMON and J_DEF (values: 'yes', 'no') give information on the demonstrativeness and definiteness of the candidate anaphor.
- I_PROPER, J_PROPER and BOTH_PROPER (values: 'yes', 'no') indicate whether a given candidate anaphor, its candidate antecedent and both are proper names.
- NUM_AGREE (values: 'yes', 'no', 'na'). The 'yes' and 'no' attribute values are used if the candidate anaphor and its candidate antecedent agree in number or disagree in number, respectively. 'Na' is used when the number of one of the constituents cannot be determined.

SYNTACTIC FEATURES

- ANA_SYNT, ANT_SYNT and BOTH_SBJ/OBJ (values: 'SBJ', 'OBJ', 'no', 'imm_prec_SBJ' (only for antecedent), 'imm_prec_OBJ' (only for antecedent). The first two features inform on the syntactic function of the candidate anaphor (ANA_SYNT) and its candidate antecedent (ANT_SYNT). If the candidate antecedent is the immediately preceding subject or object, it takes as value 'imm_prec_SBJ' or 'imm_prec_OBJ', respectively.

The BOTH_SBJ/OBJ feature checks for syntactic parallelism. For the English data sets, the three features only distinguish between subjects and objects.
For Dutch, we also integrated information on predicative complements in our feature vectors. This implies that for the three features, the additional values 'PREDC' and 'imm_prec_PREDC' are possible.

- APPOSITIVE (values: 'yes', 'no') checks whether the anaphoric NP is an apposition to the preceding NP.

STRING-MATCHING FEATURES

- COMP_MATCH (values: 'yes', 'no') is set to 'yes' in case of a complete match between the anaphor and its candidate antecedent. This feature is applied to the NP after stripping off (1) appositions, (2) postnominal prepositional modifiers (as in 'date for talks on pay-cut plan') and (3) prenominal possessive modifiers (as in 'Eastern's president').

- PART_MATCH (values: 'yes', 'no') checks for a partial match between both noun phrases. The following NP couples are examples of partially matching NPs. Whereas for English partial matching is restricted to the word level, we also performed word internal matching for Dutch. This type of matching is shown in the last two example couples in the table below. In order to do so, we used the previously described morphological analysis to split the compound words into their different parts, e.g. "pensioenspaarverzekeringen" into "pensioen+spaar+verzekeringen". These different parts were then checked for partial matching.

Candidate anaphor	Candidate antecedent
This deal	Monday's deal
the changes	top management changes
het hele conflict	het conflict over het grensgebied
die sancties	Amerikaanse economische sancties tegen beide landen
het wettelijk pensioen	het wettelijk pensioenstelsel
de pensioenverzekeringen	de pensioenspaarverzekeringen

- ALIAS (values: 'yes', 'no') indicates whether the candidate anaphor is an alias of its candidate antecedent or vice versa. The alias of a given NP is determined by removing all prepositions and determiners and then by taking the first letter of the nouns in the noun phrase. These letters are then combined in various ways. This simple approach allows us to capture the alias "IBM" which stands for "**I**nternational **B**usiness **M**achines". But

this approach fails to recognize "Pan Am" as an alias of "**Pan Am**erican Airways" or "BeCa" as an alias of "**Be**lgian **C**ockpit **A**ssociation". It also fails to recognize "MDC" as an alias of "Beweging voor Democratische Verandering", which is already a translation of the original "**M**ovement for **D**emocratic **C**hange".

- SAME_HEAD (values: 'yes', 'no') checks whether the anaphor and its candidate antecedent share the same head. Examples of NPs sharing the same head are "the board" and "National Transportation Safety Board", "the gulf" and "the Persian Gulf", "the former weapons assembly plant" and "Pantex Weapons Plant".

SEMANTIC FEATURES

English:
For the semantic features, we took into account lists with location names, male and female person names and names of organizations. Furthermore, we looked for female/male pronouns and for gender indicators such as 'Mr.', 'Mrs.' and 'Ms.'. Further information for this feature for the two English corpora was also extracted from the WordNet1.7 (Fellbaum 1998) synonyms and hypernyms. This synonym and hypernym information is provided for each different sense of the given input word, which is often ambiguous. In case of such an input word with more than one possible sense, there were two possible options.

The first option was to use word sense disambiguation (WSD) to determine the contextual meaning of a given noun (see for example Hoste et al. (2002)) and to look for a synonym for this specific meaning of the noun. Due to the rather low scores on unrestricted word sense disambiguation for English in the Senseval-2 and Senseval-3 tasks, however, we decided not to use WSD for the construction of the semantic features. For Senseval-2, for example, an official top score of 69.0% precision and recall (Mihalcea 2002) was obtained. And for Senseval-3, our own WSD system (Decadt, Hoste, Daelemans and van den Bosch 2004) outperformed all the other systems, but it only reached a top performance of 65.2% precision and recall (which was merely 2.8% better than the WordNet most frequent sense baseline). Therefore, we decided to leave the word ambiguous and we tried to exploit this ambiguity in the construction of the semantic features. We will illustrate this construction of the semantic features by means of the following WordNet1.7 entry for the ambiguous word "Washington" in the first sentence of the MUC-6 test data:

(35) "Economy: Washington, an Exchange Ally, Seems To Be Strong Candidate to Head SEC".

Sense 1
Washington, American capital, capital of the United States
=> national capital
 => capital
  => seat
   => center, centre, middle, heart, eye
    => area, country
     => region
      => **location**
       => entity
Sense 2
Washington, Evergreen State, WA
=> American state
 => state, province
  => administrative district, administrative division, territorial division
   => district, territory
    => region
     => **location**
      => entity
Sense 3
Capitol, Washington
=> federal government
 => government, authorities, regime
  => polity
   => **organization**, organisation
    => social group
     => group, grouping
Sense 4
Washington, George Washington, President Washington
=> general, full general
 => general officer
  => commissioned military officer
   => commissioned officer
    => military officer, officer
     => serviceman, military man, man, military personnel
      => skilled worker, trained worker
       => worker
        => **person**, individual, someone, somebody, mortal, human
         => organism, being, living thing
          => entity
         => causal agent, cause, causal agency
          => entity

=> President of the United States, President, Chief Executive
 => head of state, chief of state
  => representative
   => negotiator, negotiant, treater
    => communicator
     => **person**, individual, someone, somebody, mortal, human, soul
      => organism, being, living thing
       => entity
      => causal agent, cause, causal agency
       => entity
Sense 5
Washington, Booker T. Washington, Booker Taliaferro Washington
=> educator, pedagogue
 => professional, professional person
  => adult, grownup
   => **person**, individual, someone, somebody, mortal, human, soul
    => organism, being, living thing
     => entity
    => causal agent, cause, causal agency
     => entity

For the first 5 semantic features, we defined a set of semantic classes (as was also previously done in Soon et al. (2001)): "female", "male", "person", "organization", "location", "date", "time", "money", "percent" and "object". For both the anaphor and its candidate antecedent, it is checked whether they belong to one or more of these categories. If not, their value was set to 'na'. For the synonym and hypernym feature, we did not restrict ourselves to a predefined set of semantic classes and used all synonyms and hypernyms over all senses in the WordNet1.7 output.

- ANA_AMBIG, ANT_AMBIG present a concatenation of all classes the anaphor or antecedent belong to. In the previous WordNet1.7. entry, for example, the noun 'Washington' is a person, an organization and a location.
- ANA_FIRST, ANT_FIRST. This feature gives the most frequent semantic class. E.g. the noun 'Washington' is more frequently used as a location.
- SEMCLASS_AGREE (values: 'male', 'female', 'incomp', 'person', 'object', 'date', 'loc', 'no', 'na'). If the constituents are both male or both female, the value of this feature is set to 'male' and 'female', respectively. If one of the constituents is of the male gender, whereas the other constituent

is female, or vice versa, the feature is set to 'incomp'. If both NPs are persons, but when it is not possible to determine the gender of one of the NPs or of both, the feature takes as value 'person'. If both constituents are an object, a date or a location, the feature is set to 'object', 'date' and 'loc', respectively. If both NPs do not agree on one of the preceding categories, the feature value is set to 'no'. If it is not possible to determine the semantic class of one of both constituents or of both constituents, the feature takes as value 'na'. Since this feature already encapsulates gender information, we decided not to use a distinct feature for gender.

- SYNONYM (values: 'yes', 'no') looks for a noun in the anaphoric NP with the same meaning as its possible antecedent. The following pairs from the MUC-7 cross-validation data are examples of NPs which are labeled as synonymic: "the three recent crashes" and "accidents", "noise" and " a brief unidentified sound".

- HYPERNYM (values: 'yes', 'no'). A noun is a hypernym of another noun if the concept it denotes is a superconcept of the concept the other noun denotes. The following pairs from the MUC-7 cross-validation data are examples of hypernymic NPs: "the fighter" and "the aircraft", "the heavy-lift helicopter" and "the craft", "cockpit" and "that area".

- SAME_NE (values: 'I-ORG', 'I-PER', 'I-LOC', 'no'). This feature looks at the named entity type (organization, person, location) of both NPs.

Dutch:

For the extraction of the Dutch semantic features, we took into account lists with location names, organization names, person names and male and female person names. This information could then be used for the construction of the first five semantic features for the proper nouns data. For the common nouns, however, we were not able to find any resource which could provide us this type of information. As far as we know, the EuroWordNet database (http://www.illc.uva.nl/EuroWordNet/) does not have a top ontology for Dutch. The following contrastive example illustrates this lack. For both languages, the proper noun "Coca-Cola" is identified as an organization (one of the predetermined semantic classes). Furthermore, the English WordNet1.7. allows us to identify "company" as an organization. For Dutch, however, EuroWordNet cannot provide us with this information.

(36) Op de website van **Coca-Cola** is informatie te vinden over **het bedrijf** en zijn producten.

English: On the web site of **Coca-Cola** you can find information on **the company** and its products.

Lacking this type of information for the common noun NPs, we used the Celex lexical data base (Baayen, Piepenbrock and van Rijn 1993) instead to provide gender information for the head nouns of the common noun NPs. There are three basic genders in Dutch: male, female and neutral. In addition, CELEX also names female nouns which can be treated as male and nouns whose gender depends on the context in which they are used. This makes five feature values with gender information: 'male', 'female', 'neutral', 'female(male)', 'male-female'.

For the extraction of the SYNONYM and HYPERNYM feature, we used all synonyms and hypernyms in the Dutch EuroWordNet output. And finally, the last semantic feature, SAME_NE, makes use of the output of the Dutch named entity recognition output described earlier. These three semantic features can take the same values as the corresponding English features.

For our resolution system, we did not take into account discourse knowledge (e.g. information on center or focus, which is the most salient element in discourse), nor real-world knowledge.

3.2.3 The informativeness of the features in a feature vector

Having discussed and motivated our features for the disambiguation of anaphoric relations between NPs, we can now show how a feature vector will look like in all our experiments. The examples (37) and (38) show the different previously discussed features for one potential anaphor-antecedent pair. Sentence (37) and Table 3.4 show the features for the combination of the anaphor "Hall" with its candidate antecedent "NTSB Chairman Jim Hall" in the MUC-7 training data. And sentence (38) and Table 3.5 give a similar example for Dutch.

(37) **NTSB Chairman Jim Hall** is to address a briefing on the investigation in Seattle Thursday, but board spokesman Mike Benson said **Hall** isn't expected to announce any findings.

(38) **Frans Rombouts** verdwijnt als hoofd van De Post. (...) Zeker bij de Waalse socialisten was **hij** niet erg geliefd meer.
English: Frans Rombouts leaves as head of The Post. Especially among the Walloon socialists he lost popularity.

The last column in both tables represents the gain ratio values for each feature calculated on the basis of the training corpus, which is MUC-7 in case of example (37) and KNACK-2002 in case of example (38). Gain ratio (Quinlan 1993) is a feature weighting metric which calculates on the basis of the training set which features contribute most to the prediction of the class labels. It considers each feature in isolation and then measures how much information it contributes to the correct class label. In order to avoid that features with many possible values are favoured above features with fewer values, the entropy of the feature values is taken into account. Further information on this metric is given in Section 4.2.

Since the coreferential noun phrase "Hall" in (37) is a proper noun, the gain ratio values in Table 3.4 are calculated on the basis of the MUC-7 "Proper nouns" training set. Since these values are calculated on a data set only consisting of anaphoric proper nouns, the features j_pron, ij_pron, j_demon, j_proper and j_pron_i_proper evidently have a gain ratio value of zero. Furthermore, the string-matching features comp_match (GR: 0.6), part_match (GR: 0.1) and same_head (GR: 0.5), the syntactic appositive feature (GR: 0.1) and the semantic synonym feature (GR: 0.1) have the highest gain ratio values and are thus considered most informative. For the "hij" in the Dutch example sentence (38), the gain ratio values in Table 3.5 are calculated on the basis of the KNACK-2002 "Pronouns" training set. Due to the calculation of the gain ratio values on pronominal anaphors only, the features j_pron, j_def, same_ne, j_proper, both_proper, apposition, alias, synonym and hypernym have a gain ratio value of zero. Furthermore, the string-matching features and the semclass_agree feature are assigned the highest gain ratio values.

Based on the results in both tables, we could conclude that the majority of the features has a low informativeness and that the string-matching features are the most informative ones. We will come to similar findings in Section 5.2 when considering the different features in isolation. However, in the chapter on optimization by using a genetic algorithm (Chapter 6), we will show a more balanced contribution of the different features.

3.3 Summary

In this chapter we dealt with one of the crucial components in a corpus-based coreference resolution system: the selection of informative features. We first described the different preprocessing steps (tokenization, POS tagging, named entity recognition, NP chunking, relation finding and morphological analysis) taken for the construction of the English and Dutch data sets. Furthermore, we briefly motivated the use of three data sets representing the different NP types

Table 3.4: Feature vector for the combination of the anaphor "Hall" with its candidate antecedent "NTSB Chairman Jim Hall". The last column gives the gain ratio for each feature calculated on the basis of the complete MUC-7 training corpus for the proper nouns.

Feature	**value**	**gain ratio**
dist_sent:	0	0.00077483087
dist_NP:	7	0.0031589478
left_wd_3:	Mike	0.0050000493
left_wd_2:	Benson	0.0047607090
left_wd_1:	said	0.0037450397
left_pos_3:	NNP	0.0011482068
left_pos_2:	NNP	0.0012101125
left_pos_1:	VBD	0.0024301350
right_wd_1:	is	0.0033727202
right_wd_2:	n't	0.0045263081
right_wd_3:	expected	0.0044906334
right_pos_1:	VBZ	0.0020877918
right_pos_2:	RB	0.0011000606
right_pos_3:	VBN	0.0010985286
dist_lt_three:	yes	0.00084172330
j_pron:	no	0.0000000
i_pron:	no	0.012359425
ij_pron:	no	0.0000000
j_demon:	no	0.0000000
j_def:	na	0.00063950158
num_agree:	yes	0.010991740
comp_match:	no	**0.60065569**
part_match:	yes	**0.10558114**
same_ne:	I-PER	0.030398838
appositive:	no	**0.11149599**
both_proper:	yes	0.012942332
i_proper:	yes	0.012942332
j_proper:	yes	0.0000000
alias:	no	0.012940077
ana_ambig:	personobject	0.0033671661
ana_first:	person	0.0019828006
ant_ambig:	male	0.0080019086
ant_first:	male	0.0078012782
semclass_agree:	person	0.012041707
ana_synt:	SBJ	0.00082904140
ant_synt:	imm_prec_SBJ	0.011529774
both_SBJ/OBJ:	SBJ	0.0057760014
same_head:	yes	**0.53433900**
synonym:	yes	**0.11071904**
hypernym:	no	0.0058093769
j_pron_i_proper:	na	0.0000000

Table 3.5: Feature vector for the combination of the anaphor "hij" with its candidate antecedent "Frans Rombouts". The last column gives the gain ratio for each feature calculated on the basis of the complete KNACK-2002 training corpus for the pronouns.

Feature	**value**	**gain ratio**
dist_sent:	3	0.0032274788
dist_NP:	6	0.0038607636
left_wd_3:	Waalse	0.0054332164
left_wd_2:	socialisten	0.0067645112
left_wd_1:	was	0.0053498168
left_pos_3:	ADJ(prenom,basis,met-e,stan)	0.0017425291
left_pos_2:	N(soort,mv,basis)	0.0019628279
left_pos_1:	WW(pv,verl,ev)	0.0029857284
right_wd_1:	niet	0.0062851368
right_wd_2:	erg	0.0055628354
right_wd_3:	geliefd	0.0057330833
right_pos_1:	BW()	0.0026637786
right_pos_2:	ADJ(vrij,basis,zonder)	0.0022930981
right_pos_3:	ADJ(vrij,basis,zonder)	0.0018318315
dist_lt_three:	no	0.017198244
j_pron:	yes	0.000000000
i_pron:	no	0.039424517
ij_pron:	no	0.039424517
j_demon:	no	0.0057990652
j_def:	def_yes	0.000000000
num_agree:	num_yes	0.034242769
comp_match:	no	**0.14219206**
part_match:	no	**0.13856630**
same_ne:	no	0.0000000
appositive:	appo_no	0.0000000
both_proper:	no	0.0000000
i_proper:	iproper_yes	0.016957219
j_proper:	no	0.0000000
alias:	no	0.0000000
semclass_ana	male	0.020580530
semclass_ant	male	0.036198846
semclass_agree:	male	**0.11408434**
ana_synt:	SBJ	0.00088195027
ant_synt:	imm_prec_I-SBJ	0.013706274
both_SBJ/OBJ:	SBJ	0.0063553767
same_head:	no	**0.13856630**
synonym:	no	0.0000000
hypernym:	no	0.0000000
j_pron_i_proper:	yes	0.016957219

(pronouns, proper nouns and common nouns) instead of the commonly used single data set. The main motivation for this approach is that the information sources which are important for the resolution of the coreferential relations are different for each type of NP. We continued with an elaborate discussion of the information sources which can contribute to a correct resolution of coreferential relations. In this discussion, we first provided an overview of the information sources which have been used in other machine learning work of coreference resolution. And we continued with a description and illustration of the positional, contextual, morphological, lexical, syntactic and semantic features built for our experiments. We also introduced some new features, especially semantic features, such as the synonym and hypernym features.

Having built the feature vectors for our experiments, we can now continue with a description of the machine learning approaches which we will use for our experiments.

CHAPTER 4

Machine learning of coreference resolution

We will continue in this chapter with a description of the machine learners which operate on the basis of the feature vectors explained in the previous chapter.

This chapter consists of two main parts. The first three sections introduce the term 'bias' and the two machine learning packages which we will use in our experiments: the memory-based learning package TIMBL (Daelemans et al. 2002)[1], and the rule induction package RIPPER (Cohen 1995). In the second part, Section 4.4, we describe the general setup of our experiments, discuss the different classifier performance measures and we apply the two methods to the MUC-6/-7 and KNACK-2002 validation data sets.

4.1 The 'bias' of the machine learner

When performing a machine learning (of language) experiment, several factors can strongly direct the result of learning. A factor which has been studied extensively is the *bias* of the machine learner. Bias refers to the search heuristics a certain machine learning method uses and to the way it represents the learned

[1] Available from http://ilk.uvt.nl

knowledge. Decision tree learners, for example, favor compact decision trees, and ILP systems can represent hypotheses in terms of first order logic in contrast to most other learning methods which can only represent propositional hypotheses.

Theoretical studies in Machine Learning, such as the *no free lunch* theorem (Wolpert and Macready 1995), have shown that no inductive algorithm is universally better than any other: generalization performance of any inductive algorithm is zero when averaged over a uniform distribution of all possible classification problems. In order to know which learning algorithm has the right bias for language learning, it is therefore necessary to compare machine learning methods experimentally on their behavior on specific language processing tasks (see for example Mooney (1996)). The more the bias of a learning algorithm fits the properties of the task, the better its induced model will generalize to new data of the same task. A posteriori, we may be able to say something about the bias of a particular class of algorithms being suited or not for a particular class of problems. This comparative machine learning approach has gained an enormous importance with the influence of competitive NLP research evaluations such as MUC, SENSEVAL and the CoNLL shared tasks. In the MUC Message Understanding Conferences, systems were evaluated on three tasks: information extraction (the extraction of information about a specified class of events), named entity recognition (the recognition of persons, organizations, locations, etc.) and coreference resolution. The SENSEVAL competitions are organized to compare word sense disambiguation systems for different languages. The CoNLL shared tasks have already covered a variety of tasks, such as NP bracketing (the recognition of all NP structures in a text), chunking (the identification of syntactically correlated parts) and named entity recognition.

To our knowledge, this effect of 'bias' has not been investigated systematically yet in the machine learning of coreference resolution. The existing machine learning approaches to coreference resolution use the C4.5 decision tree learner (Quinlan 1993) (used by Aone and Bennett (1995), McCarthy (1996) and Soon et al. (2001)), maximum entropy learning as in Yang et al. (2003) or the RIPPER rule learner (Cohen 1995) as in Ng and Cardie (2002a;2002b;2002c). But it is by no means certain that this type of rule induction and decision tree algorithms are the most appropriate learning algorithms for a coreference resolution task. In order to determine the effect of algorithm 'bias' on learning coreference resolution, we will evaluate the performance of two completely different learning techniques on this task: *memory-based learning* and *rule induction*. Both learning methods can be described as classification-based supervised learning. Both take as input training instances consisting of feature-value pairs, which contain disambiguating information for solving the classification task, followed by the classification of that particular instance. The second column in Table 3.4 and

Table 3.5 is an example of a feature vector used in our experiments. During classification, a previously unseen test instance is presented to the learner and classified. However, these two approaches provide extremes of the *eagerness* dimension in ML (the degree in which a learning algorithm abstracts from the training data in forming a hypothesis).

The first learning approach applied to our coreferentially annotated data, is a *memory-based learning (MBL)* approach. The approach is based on the memory-based reasoning (Stanfill and Waltz 1986) and case based reasoning schemes (Riesbeck and Schank 1989, Kolodner 1993) which state that performance in real-world tasks is based on remembering past events rather than creating rules or generalizations. MBL keeps all training data in memory and at classification time, a previously unseen test example is presented to the system and its similarity to all examples in memory is computed using a similarity metric (see Section 4.2 for a discussion on similarity metrics). The class of the most similar example(s) is then used as prediction for the test instance. This strategy is often referred to as "lazy" (Aha 1997) learning.

This storage of all training instances in memory during learning, without abstracting and without eliminating noise or exceptions is the distinguishing feature of memory-based learning (MBL) in contrast with minimal-description-length-driven or "eager" ML algorithms (e.g. decision trees, rules and decision lists). *Rule induction*, which can be described as an eager learning approach, compresses the training material by extracting a limited number of rules.

The motivation for this choice of a lazy and an eager learner is that we want to investigate how both learning techniques handle the specificity of our coreferentially annotated data sets. Daelemans, van den Bosch and Zavrel (1999) showed that natural language data sets are **highly disjunctive**, which means that the data sets consist of many subregularities and buckets of exceptions. They show that a lazy learning approach is more suitable for this type of data sets because it allows extrapolation from low-frequency or exceptional cases, whereas eager methods tend to treat these as discardable noise. They show that forgetting exceptional training instances is harmful to generalization accuracy. In the machine learning literature, this is known as the *small disjuncts* problem. We will return to this in Chapter 7. A related issue, is the problem of **the highly skewed class distribution** of the training data. We will investigate how both types of classifiers cope with these imbalances. In a two-class problem, this may cause a machine learner to describe the data as one single class and treat the data from the minority class as exceptions or even noise. Or it may lead to a classifier with many small disjuncts which tends to overfit the data. In Chapter 7, we will investigate how changing or resampling the class distribution of the training data affects the classifiers ability to classify minority-class examples and majority-class examples.

We will now continue with a description of both learning techniques through their instances TIMBL and RIPPER.

4.2 TIMBL, a lazy learner

For our experiments, we used the memory-based learning algorithms implemented in TIMBL (Daelemans et al. 2002). It is an implementation of the IB1 (Aha, Kibler and Albert 1991) algorithm, with some additional features (such a different calculation of the distances between two items).

An MBL system consists of two components: a memory-based learning component and a similarity-based performance component. During learning, the learning component adds new training instances to the memory without any abstraction or restructuring. During classification, the classification of the most similar instance in memory is taken as classification for the new test instance. In other words, given a set of instances or data points in memory: (x_1, y_1) (x_2, y_2) (x_3, y_3) ... (x_n, y_n), the task at classification time is to find the closest x_i for a new data point x_q. In order to do so, the following components are crucial: (i) a distance metric, (ii) the number of nearest neighbours to look at and (iii) a strategy of how to extrapolate from the nearest neighbours.

- **a distance metric**: When presenting a new instance for classification to the MBL learner, the learner looks in its memory in order to find all instances whose input attributes are similar to the newly presented test instance. In order to do that, we have to define what is meant by similar. In other words, we need to define a *distance metric* that defines how far x_q and x_i are.

  Since our data sets only contain symbolic features, we will restrict this overview to this type of features. The most basic metric when working with patterns of symbolic features is the **overlap metric**, which is also the metric used in the TIMBL default settings. The overlap metric states that the distance between x_q and x_i is simply the sum of the differences or distance δ between the n features:

  $$\Delta(x_q, x_i) = \sum^{n}{}_{i=1} \delta(x_{qi}, x_{ii})$$

  where:

  $$\delta(x_{qi}, x_{ii}) \; = \; 0 \; if \; x_{qi} \; = \; x_{ii}$$

  $$\delta(x_{qi}, x_{ii}) \; = \; 1 \; if \; x_{qi} \neq x_{ii}$$

The overlap metric looks at the number of matching and mismatching feature values in two instances. This means that all features are considered equally important. This is the approach taken in the IB1 algorithm from Aha et al. (1991). However, IB1 does not solve the problem of modeling the difference in relevance of the various features. In most cases some features will be more informative for the prediction of the class label than others. Using the above metric in case of classification with a large number of uninformative features and a small set of informative features will strongly hinder performance. Therefore, some type of feature selection (we will return to this topic in Section 5.2) or feature weighting is required. In TIMBL, we varied the following feature weighting metrics: no weighting (as described above), information gain weighting, gain ratio weighting (default in TIMBL) and chi-squared weighting.

– We experimented with weighing each feature by **information gain**, a number expressing the average entropy reduction a feature represents when its value is known (Quinlan 1993). The information gain of a feature i is calculated as follows. Assume we have C, the set of class labels and V_i, the set of feature values for feature i. With this information, we can calculate the database information entropy. The probabilities are estimated from the relative frequencies in the training set.

$$H(C) = -\sum_{c \in C} P(c) log_2 P(c)$$

The information gain of feature i is then measured by calculating the difference in entropy between the situations with and without the information about the values of the feature:

$$w_i = H(C) - \sum_{v \in V_i} P(v) \times H(C|v)$$

– To prevent features with many possible values from being favoured above features with fewer possible values, the information gain must be normalized. Therefore, Quinlan (1993) introduced a normalized version of information gain, called **gain ratio**, which is information gain divided by split info $si(i)$, the entropy of the feature values. This is just the entropy of the database restricted to a single feature. Gain ratio is also the default feature weighting technique in TIMBL.

$$w_i = \frac{H(C) - \sum_{v \in V_i} P(v) \times H(C|v)}{si(i)}$$

$$si(i) = -\sum_{v \in V_i} P(v) log_2 P(v)$$

– Another approach, this time from statistics, is **chi-squared feature weighting** (White and Liu 1994). If we build a contingency table consisting of all classes and feature values, the χ^2 test measures the difference between the expected values and the observed values in each of the cells in this contingency table. The χ^2 statistic can be computed as follows:

$$w_i = \chi^2 = \sum_i \sum_j \frac{(E_{ij} - O_{ij})^2}{E_{ij}}$$

Where O_{ij} is the *observed* number of instances with value v_i in class c_j. E_{ij} is the number of instances that we would *expect* for n_{ij} if there is a predictive independency between feature and class (also called the 'null hypothesis'). It is calculated as follows:

$$E_{ij} = \frac{n_{.j} n_{i.}}{n_{..}}$$

Where $n_{.j}$ is the sum over column j (the classes) of the table, $n_{i.}$ is the sum over column i (the values) of the table and $n_{..}$ is the sum of all cells of the table.

Large values of χ^2 indicate that the observed values are far from expected and thus informative, whereas small values of χ^2 mean the opposite.

These feature weights obtained by information gain weighting, gain ratio weighting and chi-squared weighting can then be used in the calculation of the overlap metric, discussed above:

$$\Delta(x_q, x_i) = \sum_{i=1}^{n} w_i \delta(x_{qi}, x_{ii})$$

A second distance metric used in our experiments, besides the overlap metric, is the **Modified Value Difference Metric** (MVDM) (Stanfill and Waltz 1986, Cost and Salzberg 1993). In the overlap metric, an exact match between feature values is required. MVDM, on the other hand, is a method in which the similarity between the values of a feature is determined by looking at co-occurrence of values with target classes. The distance between two values (v_1 and v_2) of a feature is calculated as follows:

$$\delta(v_1, v_2) = \sum_{i=1}^{n} |P(C_i|v_1) - P(C_i|v_2)|$$

- **the nearest neighbours**: The nearest neighbours are the instances in memory which are near to the test item to be classified and the classification of these nearest neighbours is used as classification for the new test instance. The number of nearest neighbours is expressed by k. In the original k-nearest neighbours algorithm (Cover and Hart 1967), the k closest training examples are taken and the test instance receives the classification of the most common category among these nearest neighbours. In case of continuous feature vectors, Euclidean distance is used to calculate the similarity of two instances. In this case, it rarely happens that two nearest neighbours have the same distance. In case of discrete and symbolic features, however, for which the distance between two values is 0 if they are the same and 1 if different (the above described overlap measure), this occurs regularly. Therefore, in the TIMBL implementation of IB1, k refers to the number of nearest distances.

  This value is usually set to 1, also in TIMBL. This means that in case of $k = 1$ the instances with the nearest distance to the test instance are used for classification. In case of multiple instances at the same distance, TIMBL selects the classification with the highest frequency in the class distribution of the k-nearest distances set. In previous work (Hoste et al. 2002), we have shown that no single value of k works best for all data sets. Therefore, its value has to be determined experimentally per data set. For our parameter optimization experiments, we took 1, 3, 5, 7, 9, 11, 15, 25, 35 and 45 as possible k values. In the GA optimization experiments, the k parameter is encoded as a real value varying between 0.0 and 7.0 which represents the logarithm of the number of neighbours.

- **a model of how to extrapolate from the nearest neighbours**: The default method in TIMBL for deciding which will be the class of a new test item, is **majority voting**. This means that all nearest neighbours receive equal weight and that the most frequent class in the nearest neighbour set is taken as classification for the new test item. In this voting scheme, far neighbours are equally important as near neighbours. In order to link the choice of classification to the distance between the nearest neighbours and the new test item, **distance weighted voting** (Dudani 1976) can be used. For our experiments, we experimented with three distance weighted voting schemes: inverse distance weighting, inverse linear weighting and exponential decay weighting. In all these weighting methods, near neighbours receive a larger weight than neighbours further away from the new test item. This weighting approach can reduce the sensitivity of the system to the parameter k, though a suitable value for k must still be found.

In Section 5.4, we will show that a good choice of distance metric, weighting method, k parameter and class weight can have a large effect on performance in

TIMBL. Therefore, we strongly believe that the optimization of these parameters should be included in each machine learning experiment. We varied all these parameters for the IB1 (Aha et al. 1991) algorithm incorporated in TIMBL.

We also used a second MBL algorithm in our experiments, called IGTREE (Daelemans, van den Bosch and Weijters 1997), which has characteristics from both lazy and eager learning. It is a heuristic approximation of the computationally expensive pure IB1 k-nearest distance classifier. In IGTREE, a tree structure is created which is much smaller than the original instance base. The features are the tests in the tree and they are ordered according to one of the feature weighting methods discussed earlier. Contrary to the standard decision tree approach, IGTREE does not prune exceptional instances.

4.3 RIPPER, a greedy learner

The second learning method which we will use in our experiments is the rule learning system RIPPER, which has been developed by Cohen (1995). Like TIMBL, RIPPER requires as input an example represented as a vector of real-valued or symbolic features, followed by a class. RIPPER also allows for set-valued features, which consist of a set of atomic symbols. In a typical NLP experiment, for example, the set-valued feature can consist of a set of words. RIPPER induces classification rules on the basis of this set of preclassified examples.

Before learning, RIPPER first heuristically orders the classes. The default approach is that the classes are ordered by increasing frequency. For our task of coreference resolution, this implies that rule learning starts with the positive "minority" class. As do most decision tree learners (e.g. C4.5 (Quinlan 1993)), RIPPER uses an overfit-and-simplify learning strategy. The system is based on an earlier algorithm, IREP (Incremental Reduced Error Pruning) developed by Furnkranz and Widmer (1994). In RIPPER and IREP, rule learning proceeds as follows. First, the available training data is split into a **growing set** (RIPPER uses 2/3 of the data) and a **pruning set** (the rest of the data). Rule learning begins with an empty clause. The first step in growing a rule is to evaluate all conditions of the form $A_n == v$, where A_n represents one of the attributes given to RIPPER, and v is one of the valid values for the attribute to take. It then grows rules in a greedy fashion adding one condition at a time. RIPPER iteratively forms conjunctions of Boolean predicates which "cover" some of the positive instances of a Boolean classification while excluding all of the negative instances. In the next iteration, these positive instances are removed from the training set, and a new conjunctive clause is formed which again covers some

more positive instances while excluding all negative ones. So, during the growing phase, conditions are repeatedly added to a rule. Rule learning stops when the rule covers no more negative examples from the growing set. This, however, leads to a rule set overfitting the data. Therefore, pruning is required which implies that rule conditions are repeatedly deleted until error rate goes up. Each rule is immediately pruned by deleting any final sequence of conditions in the rule that maximize the function

$$v * (Rule, PrunePos, PruneNeg) = \frac{p-n}{p+n}$$

in which PrunePos refers to the set of positive examples in the pruning set and PruneNeg to the set of negative examples in the pruning set. p is the number of examples covered by Rule in PrunePos and n is the number of examples covered by Rule in PruneNeg. Once a rule is created, the examples covered by the rule are removed from the training data.

RIPPER continues to learn rules until the stopping condition is met. As stopping condition, a Minimum Description Length or MDL-based heuristic (see Quinlan (1995) for a description of MDL) is used for determining how many rules should be learned. The description length is obtained by adding the number of bits required to describe the classification hypothesis to the number of bits required to describe the exceptions to this hypothesis. The minimum description length principle aims to minimize this measurement which will bias the learner toward more compact rules. In RIPPER, this means that after adding each rule, the total description length of the rule set is calculated and RIPPER stops adding rules if the description length is more than d bits larger than the smallest description length found thus far. In RIPPERs default parameter settings, d is set to 64.

RIPPER also includes optimizations of the rule set. Two alternatives are considered, "replacement" and "revision". In case of replacement, a new rule is learned for each rule in the rule set, but this time pruning is done to minimize the error rate of the whole rule set on the pruning data. Revision involves revising the existing rule by growing it (instead of starting with the empty rule) and then by pruning it back. Then, MDL is used to decide whether to use the original rule, the replacement rule of the revised rule.

As a result of learning, RIPPER outputs a set of if-then rules for the "minority" class, and a default "true" rule for the remaining class. Each of these RIPPER rules have some "confidence" information: the number of matched examples (instances that conform to the rule) and the number of unmatched examples (instances that are in conflict with the rule) in the training data. Let us consider as an example the following first two rules learned for the MUC-6 "Proper

nouns" data set:

POS	1121	61	IF	a38 = same_head a22 = comp_match a36 = SBJ
POS	403	8	IF	a38 = same_head a22 = comp_match a15 = dist_lt_three

The first rule covers 1121 training examples correctly and 61 training examples incorrectly and should be read as follows: "an anaphoric proper noun is linked to its candidate antecedent if both NPs have the same head, if both NPs match completely and if they are both a subject". The second rule covers 403 examples in the training data and there are 8 instances in the training data for which the rule fails. The rule is read as follows: "an anaphoric proper noun is linked to its candidate antecedent if both NPs have the same head, if both NPs match completely and if the distance between both NPs is less than three sentences.

There are different options to RIPPER which can strongly affect learning. For our experiments reported in Section 5.4, we varied the following algorithm parameters:

- **Class ordering**. Before learning, RIPPER first decides on a heuristic ordering of the classes. There are three different ordering methods: +freq, -freq and mdl. The default class ordering is +freq. In case of +freq, the classes are ordered by increasing frequency. For our task of coreference resolution, this means that first rules are learned for the positive minority class and the final class, the majority class is then taken as default prediction. In case of -freq, the classes are ordered by decreasing frequency. In case of choosing the MDL (Minimum Description Length) option, the classes are ordered by the description length of the rule set.

- **Negative tests**. This option is used in the construction of the rules. The option is set to *-!n* (default) if the user wishes to allow negative tests in the rule conditions.

- **Hypothesis simplification**. This option allows the user to simplify RIPPER's hypothesis, which is expressed in a set of if-then rules. We varied this option between 0.5 (default), 1 and 1.5.

- **Example coverage**. With this option, the user can determine the minimal number of examples which should be covered by a rule. We varied the values of this option between 1, 2 (default), 3 and 4.

- **Loss ratio**. With this option, we can change the ratio of the cost of a false negative (a positive example which is falsely classified as being negative) to the cost of a false positive (a negative example which is falsely classified as being positive). We varied the values between 0.5, 1 (default) and 2. The general intuition is that a value below 1 will improve recall of the minority class and that a value above 1 will be beneficial for precision.

- **Optimization passes**. This option enables the user to control the number of optimization passes (see discussion above) in RIPPER's rule learning. We varied the values for this option between 0, 1 and 2 (default).

Although both learning packages provide sensible default settings, which have been validated on a number of data sets, it is by no means certain that they are the optimal parameter settings for a particular task. In Section 5.4, we will come back to these algorithm parameter settings and we will show in a comparative experiment between both learning methods the impact of algorithm parameter optimization on classifier performance. We will also show large standard deviations in the optimization experiments, which confirms the necessity of parameter optimization.

4.4 Baseline experiments

This section describes the initial experiments with the machine learning packages, TIMBL and RIPPER. We first discuss the methodology for training both classifiers and then continue with a discussion of how to evaluate their output. Furthermore, we give some initial results of the two classifiers on our task of coreference resolution.

4.4.1 Experimental setup

The general setup of our experiments is the following. For all three data sets (MUC-6, MUC-7 and KNACK-2002), we use a training set for training and a hold-out test set. For the experiments on the test set, we refer to Chapter 8. The validation experiments for TIMBL and RIPPER are performed using ***k*-fold cross-validation** (Weiss and Kulikowski 1991) on the training set. This means that the training set is split into k subsets. Iteratively, each partition is used as the hold out set while the remaining $\frac{k-1}{k}$ balance of the training data is used for training. For our experiments k was set to ten and the partitions were made at the document level.

4.4.2 Evaluation measures

In order to evaluate the results of the classifiers, some means of measuring or evaluating classifier performance is required. In some situations and for some tasks, it is important to evaluate the overall performance of the classifier. This overall performance is measured in terms of *accuracy*. Many natural language problems (such as part-of-speech tagging) are typically evaluated in terms of predictive accuracy, which considers all errors equally.

$$accuracy = \frac{number\ of\ correct\ classifications\ given\ by\ the\ system}{total\ number\ of\ test\ instances}$$

However, predictive accuracy may not be appropriate when the data set is imbalanced, as in our coreferentially annotated data set. A simple strategy of guessing the majority class would give a high predictive accuracy, without finding any coreferential chain in the data. Therefore, it is also necessary to evaluate classifier performance for the minority class. This minority class is for us the most interesting class, since it is our ultimate goal to produce a classifier that learns coreferential links. A confusion matrix can be used to lay out the different errors. Table 4.1 is an example of such a matrix. The confusion matrix lists the correct classification against the predicted classification for each class. The number of correct predictions for each class falls along the diagonal of the matrix. Since we work on a two-class problem (coreferential, non-coreferential), there are four kinds of examples after classification: true and false positives and true and false negatives. A true positive and a true negative refer to the case where respectively a positive case or a negative case is classified correctly. A false positive occurs when a negative example is classified as being positive, whereas a false negative means that a positive example is falsely classified as being negative.

Table 4.1: Confusion matrix for the task of coreference resolution.

	Predicted as coreferential	Predicted as non-coreferential
Actually coreferential	coreferential (true positive)	non-coreferential (false negative)
Actually non-coreferential	coreferential (false positive)	non-coreferential (true negative)

Since our goal is to make a classifier which can detect coreferential relations in text, we evaluated the results of our experiments in terms of precision, recall and

F-score (van Rijsbergen 1979) of the positive class. These metrics measure the ability of the classifier to classify the positive minority class examples correctly. A high recall indicates that the classifier makes few false negative errors. A high precision is obtained if the classifier produces few false positives. In the calculation of F_β, we use $\beta = 1$ in which precision and recall are considered equally important. It is however also possible to focus on recall by increasing the value of β or to emphasize precision by decreasing the value of β.

$$recall = \frac{number\ of\ correct\ anaphoric\ relations\ given\ by\ the\ system}{total\ number\ of\ anaphoric\ relations\ in\ the\ text}$$

or

$$recall = \frac{number\ of\ true\ positives}{number\ of\ true\ positives\ and\ false\ negatives}$$

$$precision = \frac{number\ of\ correct\ anaphoric\ relations\ given\ by\ the\ system}{total\ number\ of\ anaphoric\ relations\ given\ by\ the\ system}$$

or

$$precision = \frac{number\ of\ true\ positives}{number\ of\ true\ and\ false\ positives}$$

$$F_\beta = \frac{(\beta^2+1)\times precision \times recall}{\beta^2 \times (precision + recall)}$$

Another technique for evaluating classifier performance, which is becoming increasingly more popular in machine learning research are so-called ROC (Receiver Operating Characteristics) graphs (Fawcett 2003). ROC graphs are two-dimensional graphs in which the true positive rate is plotted on the Y-axis and the false positive rate on the X-axis. The main advantage of ROC graphs is that they are insensitive to class skews. The class distribution, the proportion of positive to negative instances is the relationship of the upper row (positive instances) to the lower row (negative instances). Any performance metric that uses values from both rows, such as accuracy, precision and F_β will be sensitive to skewedness of the classes. ROC graphs, on the other hand, are based on information present in a single row and therefore do not depend on class distributions.

Since the official MUC-scoring software which will be used for the test data also uses precision, recall and F_β as evaluation measures, we used these same measures and not ROC graphs for evaluation.

For the computation of the results on the ten folds, we calculated the results using both microaveraging and macroaveraging. Microaveraging and macroaveraging may give quite different results, especially if the classes are not equally distributed.

Microaveraging considers all classifications over the ten partitions as one single output and then computes accuracy, precision, recall and F_β on this concatenated output. The use of microaveraging is motivated by the unequal partitions caused by the partitioning on document level and by the unequal distribution of anaphoric relations over all documents.

Macroaveraging on the other hand, computes these scores separately for all ten sets of documents and then computes the mean of the resulting values. A motivation for the use of macroaveraging is that it allows for calculating the standard deviation or the spread around the average from the results on the single folds. Figures 4.1, 4.2 and 4.3 show the macro-averaged results of both learners on the three coreference resolution data sets. These results are obtained by applying the learning methods in their standard representation on the complete feature vector. The boxplots clearly show a large variance. The RIPPER results on the MUC-7 "Proper nouns" data, for example, show an average $F_{\beta=1}$ result of 65.72%, with a high standard deviation of 9.98.

However, for the remainder of this thesis, we will only provide the microaveraging results, in order to improve readability. For the calculation of statistical significance of the precision, recall and $F_{\beta=1}$ results we did not use the standard statistical tests (such as paired t-test), since it has been shown (Yeh 2000) that they often underestimate the statistical significance of the difference between the results. Instead, we used the computationally intensive bootstrap resampling (Noreen 1989, Yeh 2000) test to the output of the classifier. This is done by randomly drawing feature vectors with replacement (bootstrap samples) from the classifier outputs. We repeated this step 250 times. On the basis of these 250 bootstrap results, we calculated the average $F_{\beta=1}$, the standard error and the upper and lower bound of the center 90% distribution. Since we do not know if the performance of our system is distributed according to a normal distribution, the significance boundaries are determined in such a way that for 5% of the samples the $F_{\beta=1}$ rate was equal or below the lower significance boundary and that for 5% of the samples the $F_{\beta=1}$ rate was equal or above the upper significance boundary. A score X is considered to be significantly different from a score Y if score Y is not within the center 90% of the distribution of X.

4.4.3 Results on the validation data

Three baselines. Before proceeding to the classifier experiments, we first calculate three baselines.

- The **BaselineI** results in Table 4.2 are obtained by always assigning the

Figure 4.1: Macro-averaged $F_{\beta=1}$ results for Timbl and Ripper on the MUC-6 data sets

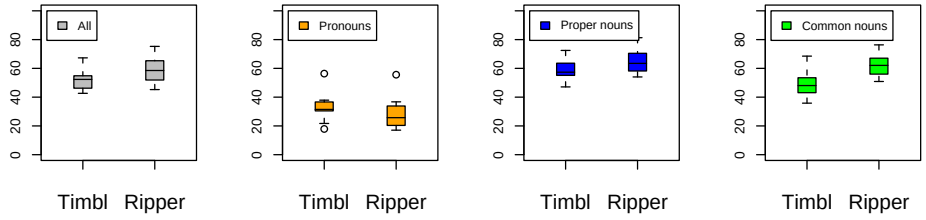


Figure 4.2: Macro-averaged $F_{\beta=1}$ results for Timbl and Ripper on the MUC-7 data sets

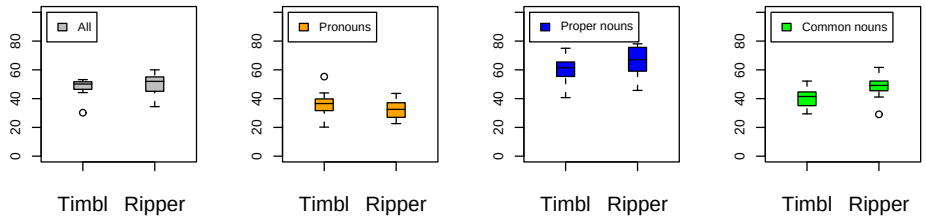


Figure 4.3: Macro-averaged $F_{\beta=1}$ results for Timbl and Ripper on the KNACK-2002 data sets

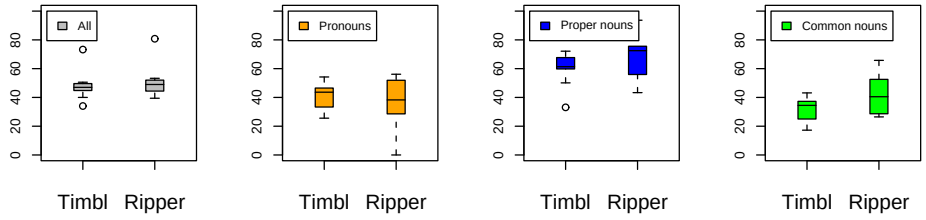


Table 4.2: A baseline score for the different data sets. The **BaselineI** results are obtained by always assigning the majority class to the instances. The **BaselineII** results are obtained by randomly assigning classes to the instances taking into account the distribution of the classes in the training set. The **BaselineIII** scores are the result of training TIMBL on a feature vector consisting of one single feature, the feature with the highest gain ratio value.

MUC-6	Acc.	Prec.	Rec.	$F_{\beta=1}$
BaselineI				
All	93.41	-	-	-
Pronouns	93.04	-	-	-
Proper nouns	92.08	-	-	-
Common nouns	95.04	-	-	-
BaselineII				
All	87.79	6.86	6.79	6.82
Pronouns	86.98	6.56	6.58	6.57
Proper nouns	85.62	7.99	7.76	7.87
Common nouns	90.66	5.51	5.48	5.49
BaselineIII				
All	95.28	71.74	46.79	56.64
Pronouns	93.04	0.00	0.00	0.00
Proper nouns	95.92	90.53	54.11	67.73
Common nouns	96.56	87.71	35.70	50.74

MUC-7	Acc.	Prec.	Rec.	$F_{\beta=1}$
BaselineI				
All	94.19	-	-	-
Pronouns	91.45	-	-	-
Proper nouns	94.00	-	-	-
Common nouns	95.73	-	-	-
BaselineII				
All	89.02	5.39	5.39	5.39
Pronouns	84.31	8.11	8.10	8.11
Proper nouns	88.62	5.24	5.24	5.24
Common nouns	91.88	4.29	4.29	4.29
BaselineIII				
All	95.41	83.12	26.25	39.90
Pronouns	92.27	61.44	25.51	36.05
Proper nouns	95.90	99.91	31.78	48.22
Common nouns	96.55	93.59	19.81	32.70

KNACK-2002	Acc.	Prec.	Rec.	$F_{\beta=1}$
BaselineI				
All	93.69	-	-	-
Pronouns	91.42	-	-	-
Proper nouns	93.82	-	-	-
Common nouns	96.08	-	-	-
BaselineII				
All	88.10	6.34	6.44	6.39
Pronouns	84.12	7.86	7.94	7.90
Proper nouns	88.16	6.88	7.31	7.09
Common nouns	92.48	4.84	4.92	4.88
BaselineIII				
All	94.47	64.73	27.15	38.25
Pronouns	90.42	27.00	6.85	10.92
Proper nouns	96.07	97.23	37.38	54.00
Common nouns	96.51	75.82	16.16	26.64

majority class to the instances. Running both classifiers without taking into account any features will lead to the same results. For TIMBL, the negative majority class will always be dominant in the nearest neighbours and will always be taken as classification for all test items. And RIPPER will not be able to learn rules for its minority class and will therefore always assign the default majority class to a given test item. We can conclude from these baseline results that accuracy scores are not a good measure for assessing the quality of a coreference resolution system. Due to the highly skewed class distribution, high accuracy scores can be obtained (above 90%) without finding any coreference, which should be the ultimate goal of all these experiments.

- Therefore, we calculated a second baseline, **BaselineII**, which takes into account the class distribution. This baseline randomly assigns classes to the test instances taking into account the distribution of the classes in the training data. The scores are obtained through a TIMBL ten-fold cross-validation experiment. The low precision, recall and $F_{\beta=1}$ results all reveal the skewedness of the classes. The resembling precision and recall scores indicate that the class distribution in the training data is similar to that of the test data.

- **BaselineIII** is inspired by the "1R" learning algorithm from Holte (1993), which computes the most informative feature and then bases its prediction on this feature alone. We followed a similar approach: the results are obtained by looking at the gain ratio values of the single features. For each data set, the feature with the highest gain ratio value is used for prediction. The scores are obtained through a TIMBL ten-fold cross-validation exper-

iment. Except for the MUC-6 and KNACK-2002 "Pronouns" data set, where even this informative feature cannot help to discriminate between positive and negative instances, all results show high precision scores.

Default classifier results. The percentages in Table 4.3 are the microaveraged results after application from TIMBL and RIPPER on the train set using ten-fold cross-validation. For both learning methods default parameter settings and the complete feature set were used. The "all" results represent the scores when training the classifiers on all available training data, whereas the "Pronouns", "Proper nouns" and "Common nouns" results show the results of the classifiers trained specifically on this NP type. The results which are reported in this and the following three chapters are validation results on the instance level. This means that per given anaphor a positive classification has to be found for each of its preceding antecedents. In chapter 8, the chapter in which the testing results are reported, only one antecedent per anaphor has to be retrieved. These results allow for comparison with similar work on the same data sets.

We can conclude the following from the default classifier results. Our hypothesis that three classifiers, each trained on one specific NP type perform better than one single classifier is confirmed for RIPPER, but not for TIMBL. The RIPPER results on the combined output of the NP type modules are always higher (MUC-7, KNACK-2002: $p<<0.01$) than the results on the data sets as a whole, whereas the TIMBL results on the combined output of the NP type modules are similar (MUC-6, KNACK-2002) or even significantly below (MUC-7: $p<<0.01$) the results on the complete data set. In the following chapters, we will investigate whether this tendency remains throughout the feature selection, parameter optimization and the joint optimization experiments.

The precision, recall and $F_{\beta=1}$ results in Table 4.3 show some clear tendencies. The highest $F_{\beta=1}$ scores are registered for RIPPER. For MUC-6, RIPPER yields a top score of 63.16% ("PPC") compared to a top TIMBL score of 56.70% ("PPC"). On the MUC-7 data, the same tendency can be observed: 51.21% (RIPPER "PPC") vs. 48.68% (TIMBL "All"). And for KNACK-2002 , RIPPER yields a top $F_{\beta=1}$ result of 51.25% ("PPC") compared to a TIMBL score of 46.78% ("All"). The lower F-scores for TIMBL are mainly caused by precision errors. The precision scores for TIMBL are up to about 30% lower than the ones for RIPPER, which implies that TIMBL falsely classifies more instances as being coreferential (false positives). RIPPER seems to be more strict in assigning a positive classification to an instance. With respect to the recall scores, the opposite tendency can be observed, but to a lesser degree: TIMBL generally obtains a higher recall than RIPPER, which implies that TIMBL produces less false negatives.

Table 4.3: Micro-averaged cross-validation results in terms of accuracy, precision, recall and $F_{\beta=1}$ after application of TIMBL and RIPPER on the complete MUC-6, MUC-7 and KNACK-2002 data sets, on the partial data sets ("Pronouns", "Proper nouns" and "Common nouns") and on the combination of the partial data sets (PPC).

	TIMBL				RIPPER			
MUC-6	Acc.	Prec.	Rec.	$F_{\beta=1}$	Acc.	Prec.	Rec.	$F_{\beta=1}$
All	94.29	56.80	55.50	56.15	96.09	84.65	49.65	**62.59**
PPC	94.35	57.19	56.21	56.70	95.98	79.73	52.59	**63.16**
Pronouns	91.88	38.33	27.42	**31.97**	93.27	54.78	19.44	28.70
Proper nouns	94.34	63.34	67.53	65.37	96.02	83.89	61.60	**71.04**
Common nouns	95.41	53.70	53.53	53.62	97.09	79.61	55.55	**65.44**

MUC-7	Acc.	Prec.	Rec.	$F_{\beta=1}$	Acc.	Prec.	Rec.	$F_{\beta=1}$
All	94.36	51.57	46.09	48.68	95.69	77.51	36.21	**49.36**
PPC	94.25	50.53	45.32	47.78	95.73	75.89	38.64	**51.21**
Pronouns	90.32	42.31	36.60	**39.25**	92.07	59.50	22.70	32.86
Proper nouns	95.35	62.36	56.87	59.49	96.58	84.58	52.56	**64.83**
Common nouns	95.23	43.06	39.17	41.03	96.79	74.56	36.76	**49.24**

KNACK-2002	Acc.	Prec.	Rec.	$F_{\beta=1}$	Acc.	Prec.	Rec.	$F_{\beta=1}$
All	93.55	48.78	44.93	**46.78**	94.93	69.49	34.92	46.49
PPC	93.66	49.75	44.90	47.20	94.99	66.34	41.75	**51.25**
Pronouns	91.44	50.11	44.81	47.31	92.76	61.08	43.14	**50.57**
Proper nouns	95.19	62.84	54.04	58.11	95.96	76.84	49.49	**60.21**
Common nouns	94.58	30.65	30.37	30.51	96.47	61.82	25.92	**36.52**

Based on the precision and $F_{\beta=1}$ results for MUC-6, MUC-7 and KNACK-2002 in Table 4.3, we could conclude that the RIPPER rule induction learner has a better 'bias' for this type of classification task than the memory-based learner TIMBL. RIPPER produces a set of rules which seeks to capture the specificity of the minority class, leading to high precision scores on all data sets. TIMBL, on the other hand, generates a large number of false positives, which is very harmful for its precision scores. One possible explanation for the large difference in precision scores is that RIPPER uses embedded feature selection for the construction of its rules. Rules are formed by greedily adding conditions to the antecedent of a rule. At each iteration, an evaluation function is used to select the feature that has the best ability to discriminate between the classes. TIMBL, on the other hand, does not use any feature selection; it performs feature weighting, but these feature weights are determined independently of each other, without

taking into account the dependencies between features. One implication of this use of feature weighting is that a large group of features with low informativeness can overrule more informative features. This is illustrated by the BaselineII results (Table 4.2): a comparison of these results with the default TIMBL results (Table 4.3) shows that classification using one single informative feature can lead to high precision scores, whereas the use of all features (even when they are weighted) generates a large number of false positives.

In the following chapter, we will further investigate this problem of feature weighting and feature subset selection for both our learning techniques. Based on the results in Table 4.3, we hypothesize that feature selection will lead to a large increase of precision scores for TIMBL. Due to the embedded feature selection in RIPPER, we do not expect that feature subset selection for RIPPER will lead to large performance improvements.

In Chapter 7, we will focus on the different recall scores and we will link the lower recall scores for RIPPER with its sensitivity to the skewed class distribution in our data sets. The problem of learning from data sets with an unbalanced class distribution occurs when the number of examples in one class is significantly larger than the number of examples in the other class. We will link this sensitivity to data imbalances to the nature of both learning approaches. In a lazy learning approach, all instances are stored in memory and no attempt is made to simplify the model by eliminating low frequency events, whereas in a eager learning approach such as RIPPER, possibly interesting information from the training data is either thrown away by pruning or made inaccessible by the eager construction of the model. For our data sets, this implies that RIPPER will prune away possibly interesting low-frequency positive data, which is harmful for its recall scores.

4.5 Summary

In this chapter, we introduced the term 'bias' and the two machine learning packages which we will use in our experiments: the memory-based learning package TIMBL and the rule induction package RIPPER. We continued the chapter with a description of the general setup of our experiments, we discussed the different performance measures (accuracy, precision, recall and F_β) and we applied the two methods to the MUC-6/-7 and KNACK-2002 validation data sets. These experiments revealed some clear tendencies. The precision scores for TIMBL are up to about 30% lower than the ones for RIPPER, whereas the opposite tendency can be observed with respect to the recall.

We suggested the different feature handling procedures in both methods as a

possible explanation for the large difference in precision scores. In the following chapter, we will further investigate this problem of feature weighting and feature subset selection for both learning techniques.

CHAPTER 5

Selecting the optimal information sources and algorithm settings

In the previous chapters we paved the way for our coreference resolution system. We constructed features which we believe to be helpful in disambiguating between coreferential and non-coreferential relations and we selected two machine learning approaches to experiment with. Furthermore, we ran an initial experiment with our coreference resolution system. In this chapter and the following chapter on genetic algorithms, we will discuss some methodological issues involved in running a machine learning (of language) experiment. We will show empirically that current methodology in comparative machine learning of language literature often leads to methodologically debatable results. In this chapter, we consider at length the importance of feature selection and the importance of the optimization of the algorithm parameters and we apply both optimization passes to our coreference resolution data sets.

5.1 There is more to it than 'bias'

In the previous chapter, we discussed the effect of the *bias* of the machine learner on its performance on our coreferentially annotated data sets. In order to know which of both learning algorithms has the right bias for this task, we compared TIMBL and RIPPER on this specific language processing task. These results revealed that the TIMBL experiments led to higher recall scores, whereas RIPPER obtained higher precision scores. Overall, RIPPER mostly outperformed TIMBL, due to the larger differences in recall scores. Since this tendency could be observed for the different data sets, we concluded from these results that RIPPER had a better 'bias' for this type of task than TIMBL.

However, comparing two or more algorithms on a specific task is complex. Apart from the algorithm bias, many other factors potentially play a role in the outcome of a (comparative) machine learning experiment:

- The **data set** used: the **sample selected** and **its size**. With respect to sample size, Banko and Brill (2001) conclude that "We have no reason to believe that any comparative conclusions drawn on one million words will hold when we finally scale up to larger training corpora". They base this point of view on experiments comparing several machine learning approaches on one typical NLP task (confusable word disambiguation in context) with data selection sizes varying from 1 million to 1 billion. Data sample size, however, is only one aspect influencing comparative results. Also the selection of high-quality training instances has an important effect on predictive accuracy (see for example Zhang (1992) and Skalak (1993,1994) for interesting work on instance and prototype selection). Class imbalances in the selected data set can also affect classification results. In a highly skewed class distribution, one class is represented by a large number of examples whereas the others are only represented by a few. In Chapter7, we will thoroughly investigate how both algorithms cope with the skewed class distribution in our coreferentially annotated data sets.

- The **information sources** used: the **features selected** for prediction, and **their representation** (e.g. binary, numeric or nominal). The presence of irrelevant features can considerably slow the rate of learning and have a negative effect on classification results. The problem of feature selection for all our data sets is extensively discussed in Section 5.2. Experimental results are given in Section 5.3.

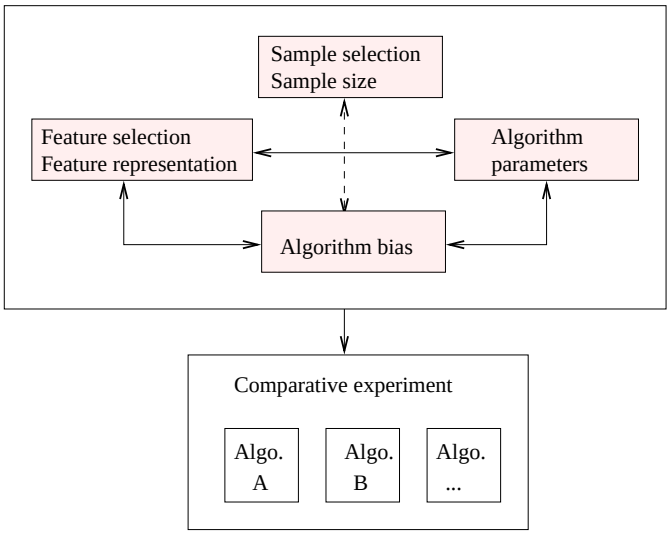
- The **algorithm parameters**. Most learning algorithms have a number of algorithm parameters which can be tuned. In Section 4.2, we gave an

overview of the algorithm parameters in TIMBL and RIPPER which can be varied. Experimental results of parameter optimization on our coreference resolution data sets are given in Section 5.4.

- These factors also interact: a feature selection which is optimal with default parameter settings is not necessarily optimal when changing the algorithm parameters. The optimal algorithm parameters for a skewed data set will not necessarily be optimal when changing the class distribution in the data set.

In Figure 5.1, a characterization of these aspects influencing a machine learning experiment is given.

Figure 5.1: Graphical representation of the aspects influencing a (comparative) machine learning experiment. The filled lines represent the experiments reported in Chapter 5. The dashed line refers to previous research on sample size of Banko and Brill (2001) and to the work on sample selection reported in Chapter 7.



In a typical comparative machine learning experiment, the impact of these factors is too often underestimated. In most comparative machine learning experiments, at least in computational linguistics, two or more algorithms are compared for a fixed sample selection, feature selection, feature representation, and (default) algorithm parameter setting over a number of trials (cross-validation), and if the measured differences are statistically significant, conclusions are drawn

about which algorithm is better suited and why (mostly in terms of algorithm bias). Sometimes different sample sizes are used to provide a learning curve, and sometimes parameters of (some of the) algorithms are optimized on training data, but this is exceptional more than common practice. This methodology has already been criticized by Banko and Brill (2001), who showed that increasing the data sample size can strongly affect comparative results. Daelemans and Hoste (2002) and Daelemans, Hoste, De Meulder and Naudts (2003) showed for different NLP data sets and UCI benchmark data (Blake and Merz 2000) that the variability recorded for the same algorithm when doing feature selection and algorithm parameter optimization is often much larger than the difference between two learning algorithms being compared. We will return to these results at the end of Chapter 6.

In the following sections, we will show for our coreference resolution data set that performance differences due to algorithm parameter optimization, feature selection, and the interaction between both also easily overwhelm the performance differences reported between algorithms in comparative experiments. We will now continue with a discussion of these factors. In Section 5.2, we give an introduction to the topic of feature selection and we discuss the informativeness of the single features for our task of coreference resolution. Section 5.3 discusses the two feature selection procedures used in the experiments: backward elimination and bidirectional hillclimbing. We also give results for both algorithms for both types of feature subset selection. Section 5.4 reports the results when doing parameter optimization for both algorithms and Chapter 6 gives a description of the joint feature selection and parameter optimization experiments using a genetic algorithm. In Chapter 7, we will extensively deal with the problem of sample selection and the highly skewed class distribution in the data.

5.2 Feature selection

In the construction of the feature vectors for our task of coreference resolution, our main goal was to build features which would help in solving our task. In the previous chapter, we ran a set of initial experiments with both our learners using the complete feature vector. One of the major observations from these experiments was that TIMBL generated a large number of false positives, which was very harmful for its precision scores, whereas RIPPER yielded precision scores up to 30% higher. We hypothesized that this discrepancy in precision scores was due to the different feature handling used by both learners: RIPPER uses embedded feature selection for the construction of its rules, whereas TIMBL performs feature weighting.

In this section, we will investigate this effect of the different feature handling through a set of feature selection experiments. We first give an introduction to the practice of feature selection and we then continue with discussion of our own experiments.

5.2.1 Filters and wrappers

The selection of relevant features and the elimination of the irrelevant features is an important problem in machine learning. There are different possible approaches to determine the informativeness of features. As shown in Section 4.2, most inductive methods incorporate some type of feature selection or feature weighting to distinguish between the informativeness of the features by assigning a real-valued weight to each feature. The weight associated with a particular feature measures its relevance in the classification task. Apart from assigning weights or degrees of informativeness to the different features, it is also possible to eliminate the non-informative features, thus creating a feature subset of the most informative features. In that case feature selection is performed as a separate process before induction. The selection of feature subsets can be considered as a special case of feature weighting. If the weights are binary valued, feature weighting becomes feature selection.

There are two main types of feature selection techniques: the filters and the wrappers. The **"filter approach"** filters out the irrelevant features before a learning algorithm is applied. A filter uses an evaluation function for determining feature relevance. The criterion for selecting the best features is independent of the performance of the learning algorithm. A possible selection criterion can be "mutual information", in which you select the k features with the highest mutual information with the class. The assumption is that features should have a strong correlation with the target class. The mutual information $I(X;Y)$ is the reduction in the uncertainty (or entropy) of X due to the knowledge of Y; it is the amount of information gained about X when Y is learned. Mutual Information is a combination of three entropy measures: the entropy of X, the entropy of Y and their joined entropy.

$$I(X;Y) = H(X) + H(Y) - H(X|Y)$$

The second approach to feature selection is called the **"wrapper approach"**. In the wrapper approach, feature informativeness is determined while running some induction algorithm on a training data set. This means that the best features are selected in relation to the problem (e.g. anaphora resolution) to be solved. The basic idea is to try different feature sets and choose the one

that gives best estimated results. The best feature selection will then be used in the learning algorithm for the real evaluation using an independent test set. Wrappers are potentially very time consuming: you have to solve the problem to be learned numerous times. For a more elaborate overview of filter and wrapper approaches, we refer to Aha and Bankert (1996), Kohavi and John (1997) and Blum and Langley (1997).

In the remainder of this section, we will discuss in more detail the informativeness of the single features and then we proceed with a discussion of the two feature selection approaches, backward elimination and bidirectional hillclimbing, both wrapper approaches, we have used in our experiments.

5.2.2 Feature informativeness

Before proceeding to the selection of the irrelevant features, we first test the informativeness of each single feature in the MUC-6, MUC-7 and KNACK-2002 training data sets. This is done on the training material as a whole and also on the three partial data sets for pronouns, proper nouns and common nouns. The informativeness of the features is determined in a 10-fold cross validation experiment by running TIMBL on each feature apart. The output of the experiments is evaluated in terms of accuracy, precision, recall and $F_{\beta=1}$. The goal of this experiment is to come to a ranking of the features according to their informativeness, a ranking which will then be used to guide the feature selection process.

The results of these experiments on the MUC-6, MUC-7 and KNACK-2002 data all reveal the same tendencies. Table 5.1 displays the results of this experiment for the MUC-6 validation material. It clearly shows the lack of informativeness of the majority of the features, when they are considered in isolation. Merely 9 out of 41 features lead to an $F_{\beta=1}$ larger than zero. Also Soon et al. (2001) observe a small number of features leading to a nonzero F-measure for both the MUC-6 and MUC-7 data sets. Among these features, the string matching features comp_match, part_match and same_head are the most informative ones for the prediction of anaphoric relations. These results are in line with the results obtained by Soon et al. (2001) and Yang et al. (2004b), who also identify the string matching feature as the most informative one. Furthermore, the semantically enriched feature semclass_agree and information about the syntactic function of the antecedent (ant_synt) also contribute to the correct prediction.

Furthermore, a comparison of the results when only taking into account the same_head feature and the default TIMBL results for the 'All' data set (Table 4.3) shows that the use of this single feature leads to a TIMBL classifier

($F_{\beta=1}$: 56.64%) outperforming the classifier using all features ($F_{\beta=1}$: 56.15%). One possible interpretation is that most features do not contribute to a correct classification (except for one). This interpretation, however, is contradicted by the RIPPER results (Table 4.3) and the RIPPER rules (Appendix B contains the RIPPER rules for the MUC-6 "Proper nouns" data set) which all reveal a combination of different features.

Since three quarters of the features leads to a zero F-measure, we used gain ratio instead (see Section 4.2 for a discussion on gain ratio) to impose an a priori ordering on the features. As expected, the highest gain ratio is given to the most informative features of Table 5.1. We will use these gain ratio values to guide the feature selection process when doing bidirectional hillclimbing. We will now continue with a description of the feature selection techniques used in the experiments.

5.3 Searching the feature space

An ideal feature vector consists of all highly informative features, which can lead the classifier to optimal performance. In Chapter 3, we described our selection of features for the task of coreference resolution. By only selecting this rather limited set of features, we already impose restrictions on the predictive power of our classifier. Given the feature vector described in that chapter, we will now discuss in this section the problem of feature subset selection. The goal is to eliminate the irrelevant features, viz. the features which add little or no additional information beyond the information provided by the other features. The system should then use the subset of features that leads to the best performance. Finding a good subset of features requires searching the space of feature subsets. However, an exhaustive search of this space is practically impossible, since this implies searching 2^n possible subsets for n attributes. So we need a more realistic approach to search the space.

In the recent existing machine learning work on coreference resolution, the importance of feature selection has been acknowledged. In Soon et al. (2001) and Ng and Cardie (2002c), efforts have been made to assess the informativeness of the features. Soon et al. (2001), for example, study the contribution of the features by training their system only taking into account one single feature and some combinations of features. And Ng and Cardie (2002c) determine feature relevance by manually omitting the features leading to low-precision rules.

For our experiments, we opted for a more systematic and verifiable feature selection approach. We use three automated techniques for the selection of the relevant features, viz. backward elimination, bidirectional hillclimbing and

Table 5.1: Informativeness of each individual feature on the complete MUC-6 train set

Feature	Accuracy	Precision	Recall	$F_{\beta=1}$
DIST_SENT	93.41%	0.00%	0.00%	0.00%
DIST_NP	93.46%	100.00%	0.69%	1.38%
DIST_LT_THREE	93.41%	0.00%	0.00%	0.00%
LEFT_WD_3	93.41%	20.00%	0.04%	0.07%
LEFT_WD_2	93.41%	0.00%	0.00%	0.00%
LEFT_WD_1	93.30%	4.63%	0.09%	0.17%
LEFT_POS_3	93.41%	0.00%	0.00%	0.00%
LEFT_POS_2	93.41%	0.00%	0.00%	0.00%
LEFT_POS_1	93.41%	0.00%	0.00%	0.00%
RIGHT_WD_1	93.41%	0.00%	0.00%	0.00%
RIGHT_WD_2	93.37%	1.27%	0.01%	0.02%
RIGHT_WD_3	93.41%	0.00%	0.00%	0.00%
RIGHT_POS_1	93.41%	0.00%	0.00%	0.00%
RIGHT_POS_2	93.41%	0.00%	0.00%	0.00%
RIGHT_POS_3	93.41%	0.00%	0.00%	0.00%
J_PRON	93.41%	0.00%	0.00%	0.00%
I_PRON	93.41%	0.00%	0.00%	0.00%
IJ_PRON	93.41%	0.00%	0.00%	0.00%
J_PRON_I_PROPER	93.41%	0.00%	0.00%	0.00%
J_DEMON	93.41%	0.00%	0.00%	0.00%
J_DEF	93.41%	0.00%	0.00%	0.00%
NUM_AGREE	93.41%	0.00%	0.00%	0.00%
I_PROPER	93.41%	0.00%	0.00%	0.00%
J_PROPER	93.41%	0.00%	0.00%	0.00%
BOTH_PROPER	93.41%	0.00%	0.00%	0.00%
ANA_SYNT	93.41%	0.00%	0.00%	0.00%
ANT_SYNT	93.67%	63.08%	9.25%	16.13%
BOTH_SBJ/OBJ	93.41%	0.00%	0.00%	0.00%
APPOSITIVE	93.41%	0.00%	0.00%	0.00%
COMP_MATCH	95.54%	81.03%	42.15%	55.46%
PART_MATCH	93.05%	46.23%	33.89%	39.11%
ALIAS	93.41%	0.00%	0.00%	0.00%
SAME_HEAD	95.28%	71.74%	46.79%	56.64%
ANA_AMBIG	93.41%	0.00%	0.00%	0.00%
ANA_FIRST	93.41%	0.00%	0.00%	0.00%
ANT_AMBIG	93.41%	0.00%	0.00%	0.00%
ANT_FIRST	93.41%	0.00%	0.00%	0.00%
SEMCLASS_AGREE	93.54%	61.33%	5.17%	9.53%
SYNONYM	93.41%	0.00%	0.00%	0.00%
HYPERNYM	93.41%	0.00%	0.00%	0.00%
SAME_NE	93.41%	0.00%	0.00%	0.00%

a genetic algorithm. These three approaches start the search at a different starting point, when searching the space of feature subsets.

One possible approach is a so-called **hillclimbing** procedure as used in forward selection, backward elimination and bidirectional hillclimbing. The idea behind hillclimbing is the following:

- Take as starting point an empty feature set, a complete feature set or a random feature set.
- Consider all neighbours of this current state by adding features (*forward selection*), removing them (*backward elimination*) or by using a combination of both techniques (*bidirectional hillclimbing*).
- Choose the feature set leading to the largest increase of performance and take this feature set as new starting point.
- Repeat the preceding two actions until no improvement can be obtained. The search is stopped when adding or removing attributes does not improve the estimate of classification accuracy.
- Return the current feature set as optimal feature set.

The main problem with this hillclimbing approach is that we are not guaranteed to find the best solution and that the search algorithm converges to a local optimum.

Contrary to hill-climbing approaches, **genetic algorithms** can allow multiple optima. In case of genetic algorithms (GAs) search does not start from a local search point, but the GA explores different areas of the search space in parallel. The search starts from a population of individuals. To decide which individual will survive into the next generation, a selection criterion is applied to determine the *fitness* of the individuals. New individuals are combined using procedures such as *mutation* and *crossover*.

We will now continue with a detailed description of the two first selection techniques used in our experiments: backward elimination and bidirectional hillclimbing. A description of the genetic algorithm used in our experiments is given in Chapter 6. All these approaches to feature subset selection are instances of the wrapper approach.

5.3.1 Backward elimination

One of the most common techniques of feature selection is backward elimination (see for example John, Kohavi and Pfleger (1994)). In case of searching the feature space with backward elimination, the search is started with the entire feature set and all features which do not make a contribution to prediction are eliminated. Figure 5.2 represents the backward elimination procedure used in our experiments. In the feature selection process, the selected features are evaluated using $F_{\beta=1}$ as evaluation criterion, since our focus is on the selection of the anaphoric relations.

Table 5.2: The backward elimination algorithm

```
begin
   F := full set of features
   while F:≠ ∅ do
     /* eliminate features */
     for f ∈ F
       set F' ← F\{f}
       train the classifier with F' and keep performance
     endfor
     set F ← F\{f*} where f* is the worst
     validation performance in for loop
     keep the validation performance with new F
   endwhile
   return best feature set F
end
```

5.3.2 Bidirectional hillclimbing

For the bidirectional hillclimbing (Caruana and Freitag 1994) procedure, we start from the gain ratio values of the features, which are calculated for the three different data sets. On the basis of the gain ratio values, uninformative features are discarded from the selection process. These uninformative features only have one feature value throughout the whole data set, and are therefore useless for prediction. The most informative features, on the other hand, are used as starting point for the selection. Features are only discarded in the "Pronouns", "Common nouns" and "Proper nouns" data sets, not in the "All" data set. Having a feature with only one feature value in the "All" data set would imply

that this feature is useless for prediction. Table 5.3 gives an overview of the discarded features and of the features used as starting point for the bidirectional hillclimbing experiments for MUC-6, MUC-7 and KNACK-2002. A general observation for all three global data sets and their partial data sets is that the string matching features "comp_match", "part_match" and "same_head" are most informative.

The features described in Table 5.3 are taken as starting point for the bidirectional hillclimbing. Contrary to the backward elimination experiments, we do not start from the complete feature set, since we want to keep processing times down. We decided to start with a feature set with all highly informative features. This allows us to limit the search space and to spend more effort to the selection procedure. In our experiments, we first examine all backward selection steps and then proceed to forward selection. Per step, the best feature selection is retained and then taken as starting point for a new selection round. When having different feature sets leading to the same top performance, the smallest feature set among them is taken as optimal feature set.

Tables 5.4 and 5.5 represent the results of TIMBL and RIPPER on the MUC-6 and MUC-7 data sets after performing backward selection and bidirectional hillclimbing. Since the tendencies observed for both data sets and also those observed for KNACK-2002 are highly similar, we decided not to list those KNACK-2002 figures. The results reveal the following tendencies:

- Based on the default scores, we saw in the previous chapter that RIPPER and TIMBL committed different types of errors. In these experiments in which both learners were applied in their default settings, RIPPER yielded much higher precision scores than TIMBL. We hypothesized that this was mainly due to the embedded feature selection in RIPPER. One major shortcoming of TIMBL is that it does not take into account dependencies between features. Based on these results, we hypothesized at the end of the previous chapter that feature selection would lead to an increase of the precision scores. This is indeed the case. Feature selection lifts the precision scores for TIMBL with up to 35%. As expected, this increase is much smaller for RIPPER due to the embedded feature selection used for the construction of the rules. We furthermore observed in the previous chapter that TIMBL yielded higher recall scores and could better distinguish false negatives from true negatives. We will come back to this issue in Chapter 7. However, globally, mainly as a consequence of the major improvements in the precision scores, TIMBL now globally outperforms RIPPER for MUC-6 ("All": $p<<0.01$). For MUC-7, the differences in F-score are not significant (Timbl "PPC": 54.24% vs. Ripper "PPC": 55.01%).

Table 5.3: Table with the discarded features and most informative features for the three global data sets and their partial data sets. Both groups of features are selected on the basis of their gain ratio values. The discarded features are not considered in the bidirectional hillclimbing procedure. The most informative features, on the other hand, are taken as starting point for the selection. Their gain ratio values are given between brackets.

Data set	discarded features
Pronouns	j_pron, j_demon, j_def, alias, synonym, hypernym
Proper nouns	j_pron, ij_pron, j_proper, j_pron_i_proper, j_dem (MUC-7)
Common nouns	j_pron, ij_pron, j_pron_i_proper, alias

	most informative features (GR > 0.099)		
Data set	MUC-6	MUC-7	KNACK-2002
Pronouns		comp_match (0.2) part_match (0.2) same_head (0.2)	comp_match (0.1) part_match (0.1) same_head (0.1) semclass_agree (0.1)
Proper nouns	comp_match (0.7) part_match (0.2) same_head (0.5)	comp_match (0.6) part_match (0.1) appositive (0.1) same_head (0.5) synonym (0.1)	comp_match (0.5) part_match (0.3) appositive (0.3) same_head (0.6)
Common nouns	comp_match (0.5) same_head (0.4)	comp_match (0.5) same_head (0.4)	comp_match (0.3) appositive (0.4) same_head (0.3)
All	comp_match (0.4) same_head (0.4)	comp_match (0.4) same_head (0.4)	comp_match (0.2) part_match (0.2) appositive (0.3) same_head (0.3) semclass_agree (0.2)

- With respect to the selection procedures, we can conclude that the use of the gain ratio scores for the selection of the informative features proves to be beneficial for the precision scores but it can also be harmful for the recall. With respect to the selection procedures, backward elimination and bidirectional hillclimbing, we cannot draw a general conclusion. For MUC-6, backward elimination performs better 8 out of 10 times for both TIMBL and RIPPER. For MUC-7, on the other hand, both methods outperform the other an equal number of times. We can conclude that a strategy combining the best of both techniques, namely bidirectional hillclimbing starting with the complete feature set would lead to the best result.

- A comparison of the "All" and "PPC" results for both algorithms reveals that training the classifiers on the different NP types and trying to capture the specificity of these single NP types through feature selection per NP type does not lead to other tendencies than those found in the default experiments. For TIMBL, no significant better results can be obtained: 64.14% ("All") vs. 63.04% ("PPC") for MUC6 and 54.01% ("All") vs. 54.24% ("PPC") on the MUC7 data. For RIPPER the best scores are obtained on the combined "Pronouns", "Proper nouns" and "Common nouns" data: 62.94% ("All") vs. 64.78% ("PPC") for MUC6 and 53.33% ("All") vs. 55.01% ("PPC") on the MUC7 data.

With respect to the selected features, we conclude that no general conclusions can be drawn. Per data set and per selection procedure, a different feature set is selected by each learning algorithm. This implies that the optimal feature selection has to be determined experimentally for each single data set. This conclusion contradicts common practice in the machine learning of coreference resolution research in which one seeks for one optimal feature vector without considering the differences in data sets. The RIPPER rules in Appendix B and the TIMBL results in Table 5.6 for the MUC-6 "Pronouns", "Proper nouns" and "Common nouns" data sets clearly exemplify these differences in feature importance between the different data sets.

Although the search for disambiguating features is central in the machine learning research for coreference resolution and for NLP tasks in general, there is no general tendency to also consider the complex interaction between all these information sources. As extensively described in Section 3.2, all coreference resolution systems use a combination of lexical, syntactic, semantic and positional features which can help to resolve coreferential relations. Recently, some initial work (e.g. Soon et al. (2001) and Ng and Cardie (2002c)) has been done to assess the informativeness of these features. Our results show that feature selection can lead to a more balanced feature vector and we therefore believe it to be indispensable in a proper NLP learning experiment.

Table 5.4: Results of TIMBL and RIPPER on the MUC-6 data sets after (i) backward selection, (ii) classification with the features with the highest gain ratio and (iii) bidirectional hillclimbing.

	TIMBL				RIPPER			
All	Acc.	Prec.	Rec.	$F_{\beta=1}$	Acc.	Prec.	Rec.	$F_{\beta=1}$
default	94.29	56.80	55.50	56.15	96.09	84.65	49.65	62.59
backward	95.73	76.38	50.98	**64.14**	96.12	84.98	49.98	**62.94**
GR	95.58	81.09	42.86	56.08	95.58	81.09	42.86	56.08
bi.hill.	95.93	77.88	53.41	63.36	95.75	79.77	47.51	59.55
PPC								
default	94.35	57.19	56.21	56.70	95.98	79.73	52.59	63.16
backward	95.42	67.24	59.33	**63.04**	96.19	82.88	53.17	**64.78**
GR	95.71	88.89	39.85	55.03	95.72	89.59	39.55	54.88
bi.hill.	96.05	84.75	48.84	61.97	96.31	88.29	50.68	64.40
Pronouns								
default	91.88	38.33	27.42	31.97	93.27	54.78	19.44	28.70
backward	92.31	43.53	35.24	**38.95**	93.57	59.25	24.43	**34.59**
GR	93.04	0.00	0.00	0.00	93.04	0.00	0.00	0.00
bi.hill.	93.68	60.86	25.97	36.41	93.86	77.19	16.70	27.46
Proper nouns								
default	94.34	63.34	67.53	65.37	96.02	83.89	61.60	71.04
backward	94.97	67.86	69.34	68.59	96.13	86.10	60.90	71.34
GR	95.97	89.46	55.65	68.62	95.98	90.22	55.19	68.49
bi.hill.	96.26	89.67	59.57	**71.58**	96.28	90.17	59.52	**71.70**
Common Nouns								
default	95.41	53.70	53.53	53.62	97.09	79.61	55.55	65.44
backward	97.23	82.42	56.12	**66.77**	97.38	85.62	56.74	**68.25**
GR	96.56	87.38	35.87	50.87	96.57	87.90	35.70	50.77
bi.hill.	96.84	85.43	43.64	57.77	97.39	87.14	55.46	67.78

Table 5.5: Results of TIMBL and RIPPER on the MUC-7 data sets after (i) backward selection, (ii) classification with the features with the highest gain ratio and (iii) bidirectional hillclimbing.

	TIMBL				RIPPER			
All	Acc.	Prec.	Rec.	$F_{\beta=1}$	Acc.	Prec.	Rec.	$F_{\beta=1}$
default	94.36	51.57	46.09	48.68	95.69	77.51	36.21	49.36
backward	95.75	73.98	41.26	52.98	95.86	77.49	40.34	53.06
GR	95.62	78.01	34.04	47.40	95.62	78.01	34.04	47.40
bi.hill.	95.84	75.39	42.08	**54.01**	95.88	77.99	40.52	**53.33**
PPC								
default	94.25	50.53	45.32	47.78	95.73	75.89	38.64	51.21
backward	94.97	59.24	45.78	51.61	95.90	76.75	43.11	**55.01**
GR	95.54	74.62	34.96	47.62	95.54	75.45	35.35	47.91
bi.hill.	95.89	77.64	42.09	**54.24**	95.90	77.76	42.43	54.85
Pronouns								
default	90.32	42.31	36.60	39.25	92.07	59.50	22.70	32.86
backward	90.99	46.86	40.89	**43.67**	92.17	59.34	26.43	36.57
GR	92.26	61.33	25.51	36.03	92.27	61.40	25.58	36.12
bi.hill.	92.27	61.55	25.51	36.07	92.39	60.74	30.94	**41.00**
Proper nouns								
default	95.35	62.36	56.87	59.49	96.58	84.58	52.56	64.83
backward	96.16	73.92	55.54	63.43	96.82	87.08	55.22	**67.59**
GR	96.07	81.02	45.09	57.94	96.08	81.41	45.01	57.97
bi.hill.	96.72	85.18	54.88	**66.75**	96.69	88.20	51.72	65.21
Common Nouns								
default	95.23	43.06	39.17	41.03	96.79	74.56	36.76	49.24
backward	95.89	52.02	39.28	44.76	96.94	76.05	40.41	**52.78**
GR	96.70	77.08	31.41	44.63	96.70	77.08	31.41	44.63
bi.hill.	96.96	78.83	38.72	**51.93**	96.94	76.77	39.70	52.33

Table 5.6: Optimal feature vector for TIMBL in all MUC-6 data sets.

Feature	Pronouns	Proper nouns	Common nouns	All All
DIST_SENT		X		
DIST_NP	X	X		
DIST_LT_THREE	X			X
LEFT_WD_3	X	X		
LEFT_WD_2	X	X		
LEFT_WD_1	X	X		
LEFT_POS_3	X	X		
LEFT_POS_2	X	X		
LEFT_POS_1				
RIGHT_WD_1		X		
RIGHT_WD_2		X		
RIGHT_WD_3		X		
RIGHT_POS_1	X			
RIGHT_POS_2	X	X		
RIGHT_POS_3				
J_PRON				X
I_PRON	X	X		
IJ_PRON	X			X
J_PRON_I_PROPER				
J_DEMON			X	
J_DEF		X	X	
NUM_AGREE	X	X		X
I_PROPER	X			
J_PROPER				X
BOTH_PROPER		X		
ANA_SYNT	X			
ANT_SYNT	X	X	X	X
BOTH_SBJ/OBJ		X		X
APPOSITIVE				
COMP_MATCH	X	X	X	X
PART_MATCH		X		
ALIAS				X
SAME_HEAD				X
ANA_AMBIG		X	X	X
ANA_FIRST		X		
ANT_AMBIG	X	X		
ANT_FIRST				
SEMCLASS_AGREE	X	X	X	X
SYNONYM		X	X	
HYPERNYM		X		
SAME_NE				

5.4 Variation of algorithm parameters

Another factor which can heavily affect performance is the optimization of the algorithm parameters. Algorithm parameter optimization is the process in which parameters of a learning system (such as learning rate for neural networks, or the number of nearest neighbors in memory-based learning) are tuned for a particular problem. Although most machine learning systems provide sensible default settings, it is by no means certain that they will be *optimal* parameter settings for some particular task.

In Sections 4.2 and 4.3, we discussed the algorithm parameters which we varied in TIMBL and RIPPER. These parameter optimization experiments are performed exhaustively over all selected parameters, which implies that 649 cross-validation runs were performed for RIPPER and 404 cross-validation runs for TIMBL. Since similar tendencies could be observed for all data sets, we will not hamper readability by providing an exhaustive overview of all percentages. Instead, we will describe and illustrate the observed tendencies.

Figure 5.2 gives a scatter plot of such a parameter optimization experiment for Ripper on the MUC-6 data. And Figure 5.3 displays the $F_{\beta=1}$ results of all algorithm parameter optimization experiments for the same MUC-6 data sets. Per algorithm and per data set, the best and worst scores are displayed, as well as the averages and deviations. Since the results of the experiments are not normally distributed, we did not calculate standard deviations. Instead, we considered our data as a normal curve, in which roughly 68% of the results are within one standard deviation of the average. This implies that we did not consider the top 16% and the bottom 16% of our data for calculating the deviations from the average.

For both figures, the following tendencies can be observed. The long vertical lines in Figure 5.3 reveal a lot of variation in the $F_{\beta=1}$ results when varying the algorithm parameters. The boxes which are mostly located in the upper area of these vertical lines indicate that the badly performing parameter combinations are in the minority. This is also clearly shown in Figure 5.2: most $F_{\beta=1}$ scores are located in the same upper area, except for the "Pronouns" data sets, where the scores reveal more variation. In the MUC-6 common nouns data set, for example, RIPPER yields an $F_{\beta=1}$ score of 17.65% for the combination of the '-freq' ordering method and the '0.5' loss ratio value. This combination proves to be highly damaging for the precision scores. Combining this below zero loss ratio value with the '+freq' or 'mdl' ordering methods, however, leads to top $F_{\beta=1}$ scores on the validation material (see Table 5.8). In Chapter 7, we will return to this loss ratio parameter and we will show that reducing the loss ratio leads to an improvement on recall and to a less restrictive set of rules for the

Figure 5.2: Scatter plot of a parameter optimization experiment for Ripper on the MUC-6 data ("All", "Pronouns", "Proper nouns" and "Common nouns"). The plot shows the F-beta scores for the four data sets for the different parameter settings (649 settings for RIPPER)

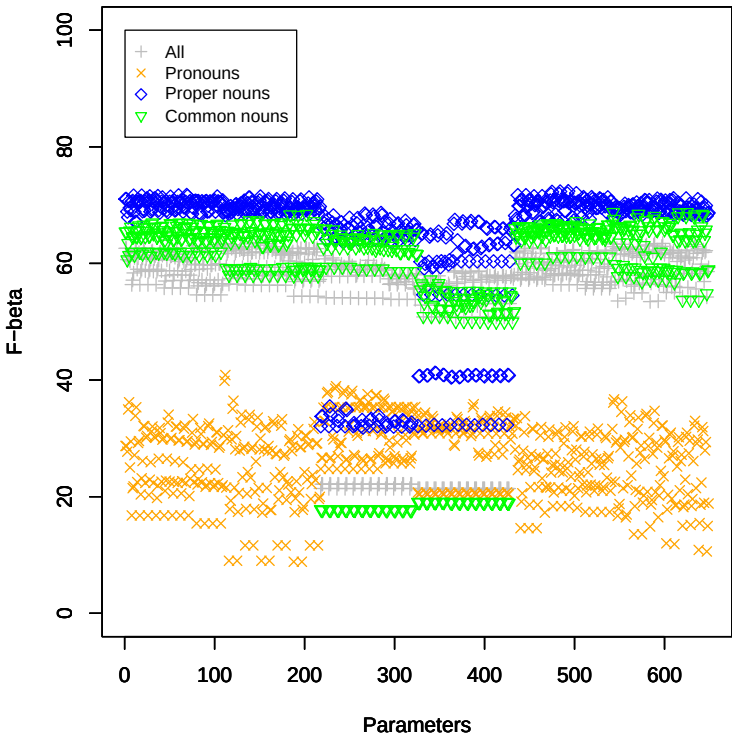
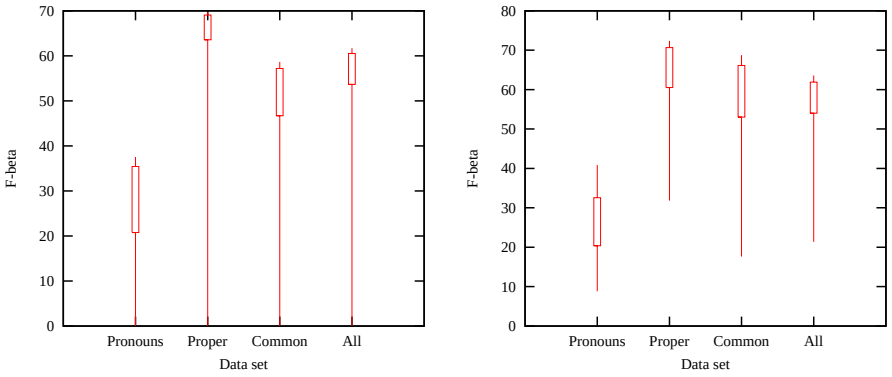


Figure 5.3: Results of TIMBL (left) and RIPPER (right) over all parameter settings for all MUC-6 data sets. The graphs show the difference between the performance obtained with the best and worst parameter settings per data set. The boxes in the graphs represent averages and deviations.



minority class. A similar conclusion can be drawn for TIMBL: combining the overlap metric with no feature weighting with high k values leads to 0.00% $F_{\beta=1}$ scores, which is not surprising given the very skewed class distribution in the validation material. The lack of distinguishing between feature values in the overlap metric often leads to more instances at the same distance. The modified value difference metric (MVDM), on the other hand, determines the similarity of the values of a feature by looking at co-occurrence of values with target classes. This implies that the nearest neighbour set will be usually much smaller for MVDM at the same value of k. As shown for the MUC-6 and MUC-7 data in Table 5.7, we can observe that the use of the Modified value difference metric in combination with no feature weighting (MVDM already has an implicit feature weighting effect), weighted class voting and a high k value gives the best results over all data sets.

A comparison of the optimal parameter settings for MUC-6 and MUC-7 data sets in Table 5.7 shows that no general conclusion can be drawn concerning these settings. Although both data sets were preprocessed in the same manner and although the division in partial data sets was identical, no setting was found to be optimal. The optimal settings merely reveal some tendencies, such as the predominant use of MVDM and weighted voting for TIMBL, and the above average use of minimal description length instance ordering and a below zero

loss ratio value for RIPPER.

Table 5.7: $F_{\beta=1}$ scores of TIMBL on the MUC-6 and MUC-7 data sets with default parameter settings and optimized parameter settings.

MUC-6	Default	Optimized	Optimized parameter settings
All	56.15	**61.70**	MVDM, no weighting, inverse linear class voting, k=45
PPC	**56.70**	61.09	combined PPC settings
Pronouns	31.97	37.53	MVDM, no weighting, inverse linear class voting, k=7
Proper nouns	65.37	69.76	MVDM, no weighting, inverse distance class voting, k=45
Common nouns	53.62	58.68	MVDM, gain ratio weighting, inverse linear class voting, k=25
MUC-7	Default	Optimized	Optimized parameter settings
All	**48.68**	**55.38**	MVDM, no weighting, inverse linear class voting, k=15
PPC	47.78	55.35	combined PPC settings
Pronouns	39.25	44.92	MVDM, chi-square weighting, inverse linear class voting, k=11
Proper nouns	59.49	65.14	MVDM, gain ratio weighting, inverse linear class voting, k=25
Common nouns	41.03	51.62	MVDM, information gain weighting, unweighted class voting, k=5

With respect to all parameter optimization experiments, we can conclude that parameter optimization overall leads to large performance increases for both learners. RIPPER, for example, which performs badly on the Pronouns data sets when using the default parameter settings (MUC-6: 28.70%, MUC-7: 32.86%), can considerably increase its scores on these data sets after optimization of its parameters (MUC-6: 40.86%, MUC-7: 46.86%). Furthermore, we observed that parameters cannot be generalized. Optimal parameter settings for one data set cannot be generalized to other similar data sets. These results confirm earlier findings reported in Hoste et al. (2002), Daelemans and Hoste (2002), Daelemans, Hoste, De Meulder and Naudts (2003) and Decadt et al. (2004), where we observed for a number of NLP data sets (word sense disambiguation, the prediction of the diminutive suffix in Dutch, part-of-speech tagging) and for some UCI machine learning data sets (Blake and Merz 2000) that changing the algorithm parameter settings can have great effects on classifier performance. We will return to these experiments in the following chapter.

Table 5.8: $F_{\beta=1}$ scores of RIPPER on the MUC-6 and MUC-7 data sets with default parameter settings and optimized parameter settings.

MUC-6	Default	Optimized	Optimized parameter settings
All	62.59	63.59	+freq, disallow negative tests, hypothesis simplification=1, loss ratio=0.5, force=1, optimization passes=2
PPC	**63.16**	**65.72**	combined PPC settings
Pronouns	28.70	40.86	+freq, disallow negative tests, hypothesis simplification=0.5, loss ratio=0.5, force=1, optimization passes=2
Proper nouns	71.04	72.39	mdl, allow negative tests, hypothesis simplification=1, loss ratio=1, force=3, optimization passes=2
Common nouns	65.44	68.78	mdl, disallow negative tests, hypothesis simplification=0.5, loss ratio=0.5, force=1, optimization passes=1
MUC-7	Default	Optimized	Optimized parameter settings
All	49.36	54.56	mdl, disallow negative tests, hypothesis simplification=1, loss ratio=0.5, force=2, optimization passes=2
PPC	**51.21**	**55.97**	combined PPC settings
Pronouns	32.86	46.86	-freq, allow negative tests, hypothesis simplification=1.5, loss ratio=2, force=4, optimization passes=0
Proper nouns	64.83	66.13	mdl, disallow negative tests, hypothesis simplification=0.5, loss ratio=1, force=1, optimization passes=2
Common nouns	49.24	51.25	mdl, allow negative tests, hypothesis simplification=1.5, loss ratio=0.5, force=4, optimization passes=2

5.5 Summary: the need for combined optimization

Based on the results in Sections 5.3 and 5.4, we can state that changing any of the architectural variables (such as algorithm parameters and information sources) can have great effects on the performance of a learning method, making questionable many conclusions in the literature based on default settings of algorithms or on limited optimization. Two main conclusions can be drawn from the experiments in this chapter. (i) In the feature selection experiments, we could observe the large effect feature selection can have on classifier performance. Especially TIMBL seemed to be very sensitive to a good feature subset. We could also observe for all data sets that the feature selection considered to be optimal for TIMBL could be different from the one optimal for RIPPER. (ii) Furthermore, in the parameter optimization experiments, we observed that the 'vertical' performance differences are much larger than the 'horizontal' algorithm-comparing performance differences. The fact that we could observe large deviations in the optimization experiments, also confirms the necessity of parameter optimization.

It is our impression that these effects are at least intuitively known by most researchers in the ML of natural language field, but little grounded evidence or explanations are available in the literature. Moreover, there appears to be little understanding of the **interaction** between these variables. Many empirical findings, though illustrative, are observations on experiments in which one or two variables are alternated, but in which no overall optimization of parameters, architecture and feature representation is undertaken (Mooney (1996), Escudero, Marquez and Rigau (2000), Ng and Lee (1996), Lee and Ng (2002) and others). These studies explore only a few points in the space of possible experiments for each algorithm to be compared. The experiments in this chapter and the experiments in the following chapter, in which we perform combined feature selection and parameter optimization, show that there is a high risk that other areas in the experimental space may lead to radically different results and conclusions. In general, the more effort is put in optimization (through feature selection, parameter optimization and their joint optimization), the more reliable the results and the comparison will be.

In the following chapter, we proceed to a next optimization step in a set of experiments exploring the interaction between feature selection and parameter optimization. Given the combinatorially explosive character of this type of optimization, we have chosen for genetic algorithms as a computationally feasible way to achieve this.

CHAPTER 6

Genetic algorithms for optimization

In the previous chapter, we showed that a proper comparative experiment requires extensive optimization and that the performance increase obtained by this optimization is considerable. In the feature selection experiments, we could observe the large effect feature selection can have on classifier performance. And in the parameter optimization experiments, we observed large deviations which confirm the necessity of parameter optimization. In these previous experiments, we explored feature selection while keeping the parameters constant and we explored parameter optimization while keeping the feature vector unchanged. We did not consider the interaction between feature selection and parameter optimization.

We will now proceed to a next optimization step in a set of experiments performing joint feature selection and parameter optimization. Joint feature selection and parameter optimization is essentially an optimization problem which involves searching the space of all possible feature subsets and parameter settings to identify the combination that is optimal or near-optimal. Due to the combinatorially explosive nature of this type of experiment, a computationally feasible way of optimization has to be found. This chapter investigates the use of a wrapper-based approach to feature selection using a genetic algorithm in conjunction with our two learning methods, TIMBL and RIPPER. In Section 6.1,

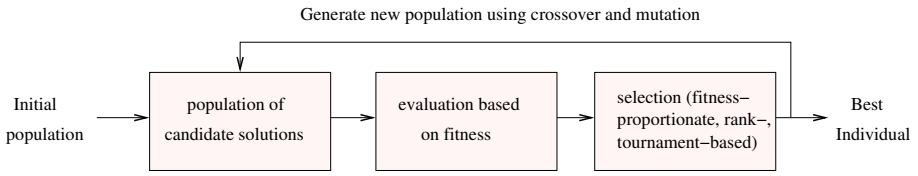
we give an introduction to genetic algorithms. Section 6.2 discusses the implementation details for running the experiments and gives experimental results on the three data sets. We conclude this chapter with a summary and discussion.

6.1 Genetic algorithms

Since exhaustive search in large search spaces is computationally not feasible in practice, genetic algorithms have for a long time been an attractive approach to find optimal or near-optimal solutions. Standard references in GA-literature include Goldberg (1989), Holland (1975), Michalewicz (1992) and Mitchell (1996).

Genetic algorithms are search methods, based on the mechanics of natural selection and genetics. They require two things: Darwinian fitness-based selection and diversity. Central principles in genetic algorithms are **selection**, **recombination** and **mutation**. As illustrated in Figure 6.1, the principle behind GAs is quite simple: search starts from a population of individuals, which all represent a candidate solution to the optimization problem to be solved. Applied to our data set, the problem to be solved will be joint parameter optimization and feature selection. These individuals are typically represented as a bit string of fixed length, called a "chromosome" or "genome". The chromosomes can be any data structure: real numbers, lists of rules, program elements, bit strings, etc. In our experiments, the individuals are represented as bit strings. Each individual contains particular values for all algorithm parameters and for the selection of the features. A possible value of a bit is called an allele. The population of chromosomes has a predefined size. Larger population sizes increase the amount of variation present in the population at the expense of requiring more fitness function evaluations.

Figure 6.1: Graphical representation of an optimization procedure using a genetic algorithm.



Fitness-based selection To decide which individuals will survive into the next generation a selection criterion is applied defining how good the individual

is at solving the problem, its **fitness**. In the feature subset selection problem, for example, the fitness function would evaluate the selected features with respect to the classification accuracy of the classifier. After the fitness assignment, the **selection** process selects the fitter individuals to produce offspring for the next generation. Some of the most common selection techniques are proportional or roulette wheel selection (Goldberg 1989), tournament-based selection (Goldberg and Deb 1991) and truncation selection (Crow and Kimura 1970). In case of *proportional selection or roulette wheel selection*, the selection probability of an individual is determined by its fitness divided by the sum of fitnesses. The individuals are mapped to segments of a line and each individual receives a segment size which is proportional to its fitness. A random number is generated and the individual whose segment spans this number is selected. This procedure is repeated until the desired number of individuals is selected. In *truncation selection*, individuals are sorted according to their fitness and only the best individuals, i.e. the individuals above a user defined truncation threshold, are uniformly selected as parents. In *tournament selection*, each time an individual is selected by randomly drawing a predefined number (mostly two) of individuals from the population. The best individual from this group is selected as parent. This process is repeated as often as individuals to choose.

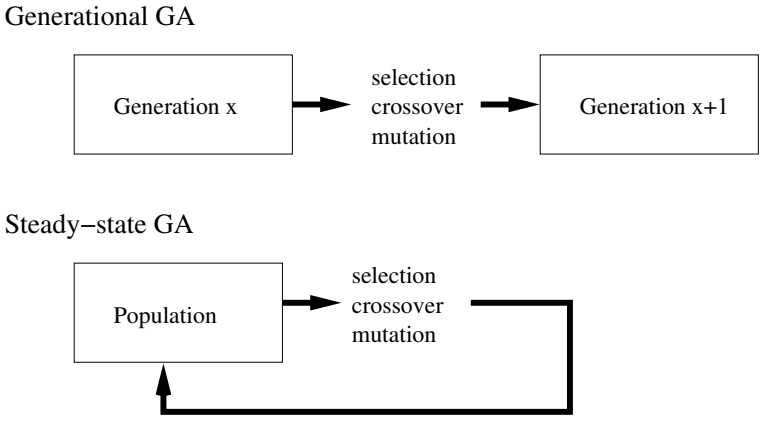
Mutation and crossover In order to combine effective solutions and maintain diversity in the population, chromosomes are combined or mutated to breed new individuals. The **mutation** operator forms a new chromosome by making alterations to the information contained in the genome of a parent according to a given probability distribution, expressed in the mutation rate. Large mutation rates increase the probability of destroying a good chromosome, but prevent premature convergence. Depending on the type of genes, different mutation strategies can be used. In case of bit strings, this mutation is realized by random negation of single bits. In case of real-valued genes, mutation is performed by adding random noise. Often, a Gaussian distribution is used.

Crossover is an operator which creates an offspring's chromosome by joining segments chosen alternately from each of two parents' chromosomes which are of fixed length. This crossover reproduction is performed with a certain probability: the crossover rate. This crossover rate can vary between 0 (no crossover) and 1 (crossover always applies). A high crossover rate is used to encourage good mixing of the chromosomes. For most applications a crossover rate of 0.75 to 0.95 is employed. The combination of parent chromosomes is usually made by selecting one or more crossover points, which split the chromosome in different portions. There a three basic crossover types: one-point, two-point and uniform. In case of one-point crossover, both parents are split at a selected point and a new chromosome is created by combining the parts of each of the

two parent chromosomes. In case of two-point crossover, two crossover points are selected in the parent chromosomes and the three portions are linked to compose a new chromosome. Uniform crossover operates on the bit level. For each bit, it is randomly decided, whether it will survive in the next generation. One-point and two-point crossover produce two offspring, while uniform crossover produces only one.

Generational vs. steady-state There are two main types of genetic algorithms which differ in the way they replace the old population: the generational or synchronous genetic algorithms and the steady-state or asynchronous genetic algorithms. A graphical representation of both types of genetic algorithms is provided in Figure 6.2. In a generational GA, the fittest individuals from the old population are selected and a new population is generated through the use of mutation and crossover. The whole old population is replaced simultaneously by the offspring population at the end of each generation. In an asynchronous GA, on the other hand, the fittest individuals are selected and a number of offspring are created and used for replacing the weakest individuals in the population. The individuals of the population are replaced by their offspring sequentially during the generation.

Figure 6.2: Graphical representation of a generational and a steady-state genetic algorithm.



Genetic algorithms can provide an alternative to the more classical search and optimization methods. As shown in the previous chapter, classical methods such as forward selection and backward elimination can get stuck in local optima. One of the advantages of genetic algorithms is that they do not start

from a local search point but explore different areas of the search space in parallel. Crossover and mutation implement a pseudo-random walk through the search space of all possible solutions. The walk is random because crossover and mutation are unbiased, non-deterministic. The walk is pseudo-random because the genetic algorithm aims to maximize the quality of the solutions using a fitness function. However, genetic algorithms cannot guarantee that they will find an optimal solution. But they remain an attractive approach to finding near-optimal solutions.

6.2 A genetic algorithm approach for feature selection and parameter optimization

6.2.1 Experimental setup

For our experiments, we used a genetic algorithm to perform joint feature selection and parameter optimization (see Kool, Daelemans and Zavrel (2000) and Kool, Zavrel and Daelemans (2000) for a similar approach). In the previous chapter, we looked at the large search space we are confronted with from two different perspectives. We first used heuristics for feature selection in an attempt to find the optimal feature vector for the solution of our anaphora resolution problem. Heuristics were used since exhaustive feature selection was computationally very expensive. However, both backward and forward selection procedures are optimal at each stage, but are unable to anticipate complex interactions between features that might affect the performance of the classifier. In the parameter optimization experiments, we kept the feature vector constant and tested the performance of the classifiers with different parameter settings. In these experiments, we did not take into account possible interactions between the choice of features and the choice of parameters. Joint feature selection and parameter optimization involves searching the space of all possible feature subsets and parameter settings to identify the combination that is optimal of near-optimal. This optimal combination cannot be found using exhaustive search, since this is in practice computationally unfeasible for the large search spaces we are confronted with. We therefore used a genetic algorithm.

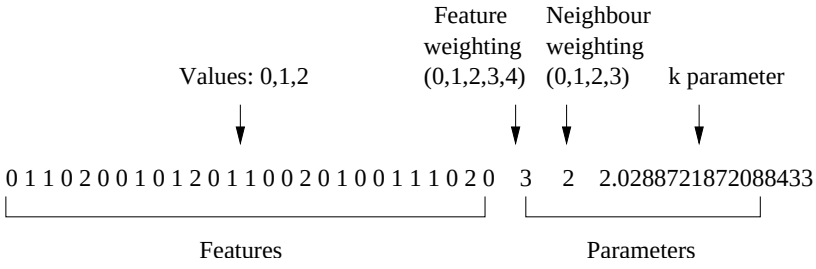
DeGA For the experiments we used a generational genetic algorithm implemented in the DeGA ("Distributed Evaluation Genetic Algorithm") framework[1]. The DeGA is a generic framework written in Java for evolutionary algorithms with evaluations distributed over a cluster of computers. It uses the

[1] More information on DeGA can be found at http://www.islab.ua.ac.be/software

Sun GridEngine queuing system and Java RMI. It features an evaluation cache, a generational GA and an implementation of the CMA-ES (Covariance Matrix Adaptation Evolution Strategy). We used the generational genetic algorithm in its standard representation in combination with our two classification techniques TIMBL and RIPPER. We are aware that the optimization problem we are trying to solve with genetic algorithms also applies to the GAs themselves. Therefore, we decided to use the generational GA implemented in DeGA in its default settings. The following GA parameters were used and kept constant:

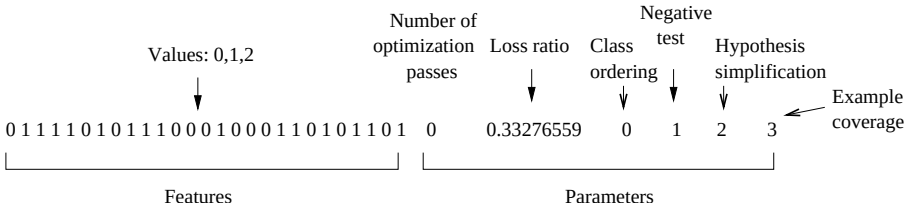
maximal number of generations	30
population size	10
crossover type	uniform crossover
crossover rate	0.9
selection type	tournament selection
selection size	2
mutation type	discrete, Gaussian (k and loss ratio)
discrete mutation rate	0.02

Representing the individuals In our experiments the individuals are represented as bit strings. Each individual contains particular values for all algorithm parameters and for the selection of the features. For TIMBL, the large majority of these features control the use of a feature (ignore, weighted overlap, modified value difference metric) and are encoded in the chromosome as ternary alleles. At the end of the chromosome, the 5-valued feature weighting parameter and the 4-valued distance weighting parameter are encoded, together with the k parameter which controls the number of neighbours. The latter is encoded as a real value which represents the logarithm of k value. The following is an example individual representing one TIMBL experiment:



A similar approach is followed for encoding the RIPPER parameters into an individual. The features are encoded as binary alleles. At the end of the chromosome, the different algorithm parameters are represented (see 4.3): the three-valued class ordering parameter (+freq, -freq or mdl), the two-valued negative

tests parameter (-!n or nothing), the three-valued hypothesis simplification parameter (0.5, 1 or 1.5), the four-valued example coverage parameter (1, 2, 3 or 4), the three-valued parameter expressing the number of optimization passes (0, 1 or 2). The loss ratio parameter is encoded as a real value, varying between 0 and 1.



For the features and the parameters in both the TIMBL and RIPPER individuals, except for the k and loss ratio parameters, discrete mutation is used, which randomly chooses another value for the feature/parameter. In case of the k and the loss ratio parameters, Gaussian noise is added to the current value.

$F_{\beta=1}$ as fitness function To decide which individuals will survive into the next generation, the $F_{\beta=1}$ result returned by TIMBL or RIPPER is used as fitness function. The genetic algorithm in DeGA also allows the use of an initialization file, in which you store individuals of which you already know that they perform well. For all experiments, we filled the initialization file with the ten best results obtained in the feature selection and parameter optimization experiments. This use of an initialization file helps us to speed up the GA optimization process.

6.2.2 Experimental results

The GA optimization experiments confirm the tendencies observed in the optimization experiments in the previous chapter. Table 6.1 shows the following:

Feature selection (previous chapter), parameter optimization (previous chapter) and their joint optimization can cause large variation in the results of both classifiers. All three optimization steps lead to a large improvement on the default results. In the default experiments, we could observe some clear tendencies with respect to the precision and recall scores. We saw that the precision scores for TIMBL were up to about 30% lower than the ones for RIPPER, which implies that TIMBL falsely classifies more instances as being coreferential (false positives). Furthermore, with respect to the recall scores, the opposite tendency could be observed, but to a lesser degree: TIMBL generally obtained a higher recall than RIPPER, which implies that TIMBL produces less false negatives. The

Table 6.1: Cross-validation results in terms of precision, recall and $F_{\beta=1}$ of TIMBL and RIPPER on the complete MUC-6, MUC-7 and KNACK-2002 data sets, on the partial data sets ("Pronouns", "Proper nouns" and "Common nouns") and on the combined output of the partial learners ("PPC"). Columns 2-4 provide the results of both learners without feature selection and in their default parameter settings. Columns 5-7 give the results after joint feature selection and parameter optimization using a genetic algorithm.

MUC-6	Default			GA optimization		
TIMBL	Prec.	Rec.	$F_{\beta=1}$	Prec.	Rec.	$F_{\beta=1}$
All	56.80	**55.50**	56.15	**83.22**	52.17	**64.14**
PPC	57.19	**56.21**	56.70	**79.13**	54.27	**64.38**
Pronouns	38.33	27.42	31.97	**45.73**	**38.48**	**41.80**
Proper nouns	63.34	**67.53**	65.37	**88.92**	59.57	**71.34**
Common nouns	53.70	53.53	53.62	**87.58**	**54.39**	**67.11**
RIPPER	Prec.	Rec.	$F_{\beta=1}$	Prec.	Rec.	$F_{\beta=1}$
All	**84.65**	49.65	62.59	73.66	**57.36**	**64.49**
PPC	**79.73**	52.59	63.16	76.01	**59.04**	**66.46**
Pronouns	**54.78**	19.44	28.70	50.84	**34.60**	**41.17**
Proper nouns	83.89	61.60	71.04	**86.03**	**62.84**	**72.63**
Common nouns	**79.61**	55.55	65.44	73.12	**66.98**	**69.92**

MUC-7	Default			GA optimization		
TIMBL	Prec.	Rec.	$F_{\beta=1}$	Prec.	Rec.	$F_{\beta=1}$
All	51.57	**46.09**	48.68	**75.29**	45.24	**56.52**
PPC	50.53	**45.32**	47.78	**76.45**	45.23	**56.84**
Pronouns	42.31	36.60	39.25	**61.28**	**38.96**	**47.64**
Proper nouns	62.36	**56.87**	59.49	**85.92**	55.11	**67.15**
Common nouns	43.06	**39.17**	41.03	**80.45**	38.76	**52.31**
RIPPER	Prec.	Rec.	$F_{\beta=1}$	Prec.	Rec.	$F_{\beta=1}$
All	**77.51**	36.21	49.36	67.19	**48.27**	**56.18**
PPC	**75.89**	38.64	51.21	67.98	**49.54**	**57.31**
Pronouns	**59.50**	22.70	32.86	49.74	**49.61**	**49.68**
Proper nouns	84.58	52.56	64.83	**87.05**	**55.25**	**67.60**
Common nouns	**74.56**	36.76	49.24	72.80	**42.03**	**53.30**

KNACK-2002	Default			GA optimization		
TIMBL	Prec.	Rec.	$F_{\beta=1}$	Prec.	Rec.	$F_{\beta=1}$
All	48.78	44.93	46.78	**71.83**	**45.50**	**55.71**
PPC	49.75	44.90	47.20	**70.22**	**49.74**	**58.24**
Pronouns	50.11	44.81	47.31	**67.65**	**53.04**	**59.46**
Proper nouns	62.84	54.04	58.11	**80.07**	**54.87**	**65.11**
Common nouns	30.65	30.37	30.51	**59.58**	**33.49**	**42.88**
RIPPER	Prec.	Rec.	$F_{\beta=1}$	Prec.	Rec.	$F_{\beta=1}$
All	**69.49**	34.92	46.49	61.51	**61.93**	**61.72**
PPC	**66.34**	41.75	51.25	60.68	**62.26**	**61.46**
Pronouns	**61.08**	43.14	50.57	58.95	**69.69**	**63.87**
Proper nouns	**76.84**	49.49	60.21	69.36	**62.71**	**65.87**
Common nouns	**61.82**	25.92	36.52	51.57	**43.48**	**47.18**

bold results in Table 6.1 show that optimization mainly wipes out these weaknesses: the increase of $F_{\beta=1}$ scores for TIMBL is mainly obtained through a large increase of precision scores for TIMBL (sometimes at the cost of recall), whereas the increase of $F_{\beta=1}$ scores for RIPPER is mainly due to the increase of recall scores (sometimes at the cost of precision).

Furthermore, we can observe that the performance differences inside one single learning method can be much larger than the method-comparing performance differences. The application of TIMBL and RIPPER on the MUC-6 "Pronouns" data set, for example, leads to default $F_{\beta=1}$ scores of 31.97% and 28.70% respectively and a 3% performance difference. But optimization within one single algorithm, for example TIMBL, leads to a performance increase of about 10% (from 31.97% to 41.80%). The TIMBL and RIPPER results on the MUC-7 "Common nouns" data set are another illustrative example. In their default representation, TIMBL and RIPPER yield a 41.03% and a 49.24% $F_{\beta=1}$ score, respectively. Optimization leads to a large performance improvement and to a less prominent performance difference: 52.31% for TIMBL and 53.29% for RIPPER. In conclusion, we can state that we cannot draw conclusions of one classifier being better on a particular task than another classifier, when only taking into account default settings or limited optimization.

In order to determine which of both learning techniques performs better on the task of coreference resolution, we applied a bootstrap resampling test to estimate significance thresholds. This test was done on the optimized "All" and "PPC" results of both learners and reveals that for half of the results none of both learners significantly outperforms the other learner (MUC-6 and MUC-7 "All") and that for the other half RIPPER significantly outperforms TIMBL (MUC-6 "PPC" and KNACK-2002 "All" and "PPC"). These results which do not reveal a clear supremacy of one learner over the other confirm the necessity

of optimization.

With respect to the use of three classifiers, each trained on the coreferential relations of a specific type of NP, instead of one single classifier covering all coreferential relations, we could observe in the default experiments that the RIPPER results on the combined output of the NP type learners were always higher (MUC7, KNACK-2002: $p<<0.01$) than the results on the data sets as a whole, whereas the TIMBL results on the combined data sets were similar (MUC-6, KNACK-2002) or even significantly below (MUC-7: $p<<0.01$) the results on the complete data set. However, after optimization this tendency becomes less clear. A comparison of the "All" and "PPC" results shows that three classifiers, each trained on one specific NP type perform better than one single classifier in 5 out of 6 cases (not for the Ripper results on KNACK-2002). But this difference in performance is only significant in 3 out of 6 cases (for TIMBL on KNACK-2002 and RIPPER on MUC-6 and MUC-7). In short, we can conclude that no convincing evidence is found for our initial hypothesis that three more specialized classifiers, each trained on the coreferential relations of a specific type of NP will perform better on the task of coreference resolution than one single classifier covering all coreferential relations.

6.3 The optimal features and parameter settings

In order to determine whether any general observations could be made concerning the optimal features and parameter settings for our coreference resolution task, we took the optimal results of each GA experiment as starting point and then applied a bootstrap resampling test on the output of the best GA individual. This bootstrap resampling test was repeated 250 times and gave us an average $F_{\beta=1}$ result of these 250 populations and a standard deviation. For GA individuals whose $F_{\beta=1}$ scores were between the two significance boundaries, we investigated whether general conclusions could be drawn with respect to the selected features and parameters. With respect to the selected features, we looked for general tendencies independently of the type of data set (Figure 6.3) and inside one type of data sets (Figures 6.4, 6.5 and 6.6).

With respect to the selected features, the following general observations can be made:

- All figures (6.3, 6.4, 6.5 and 6.6) show that RIPPER selects fewer features than TIMBL. This can be explained through the different feature handling in both learning techniques. For RIPPER, a feature is either on or off. For TIMBL, a feature is either on, off or MVDM, which implies that no

Figure 6.3: Number of times a feature is selected in all optimal GA individuals.

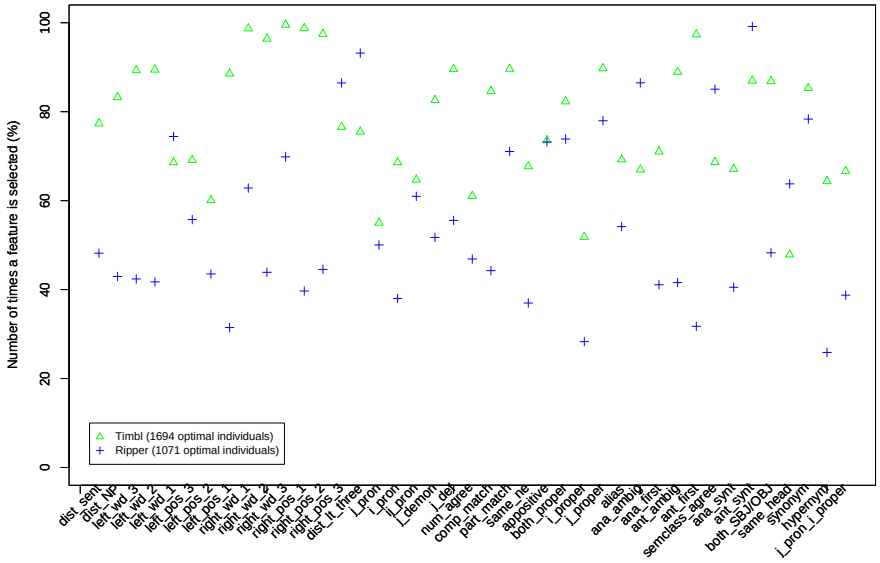


Figure 6.4: Number of times a feature is selected in the individuals for the "Pronouns" data.

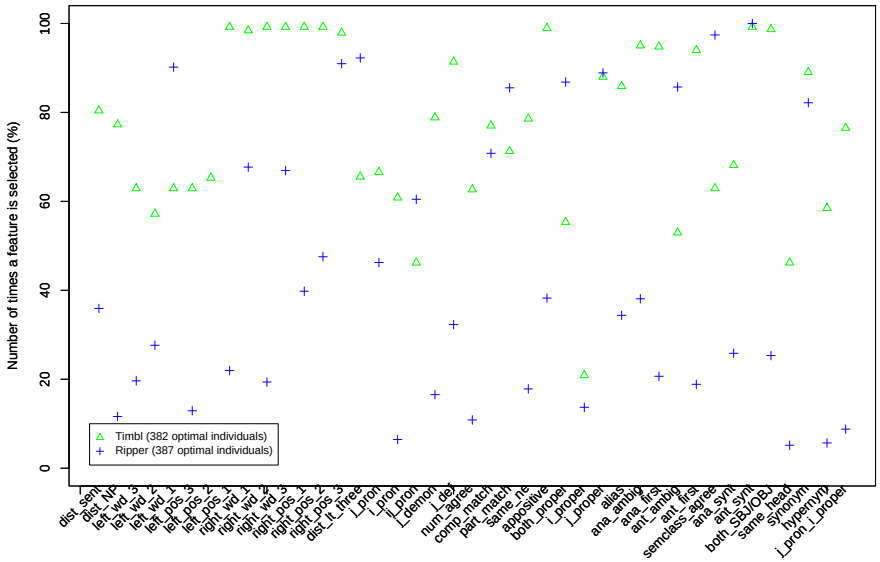


Figure 6.5: Number of times a feature is selected in the individuals for the "Proper nouns" data.

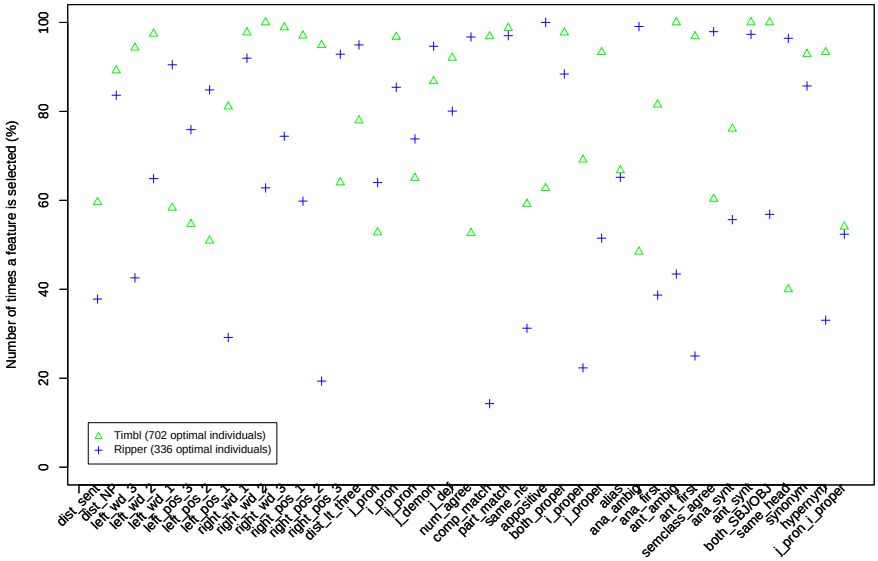
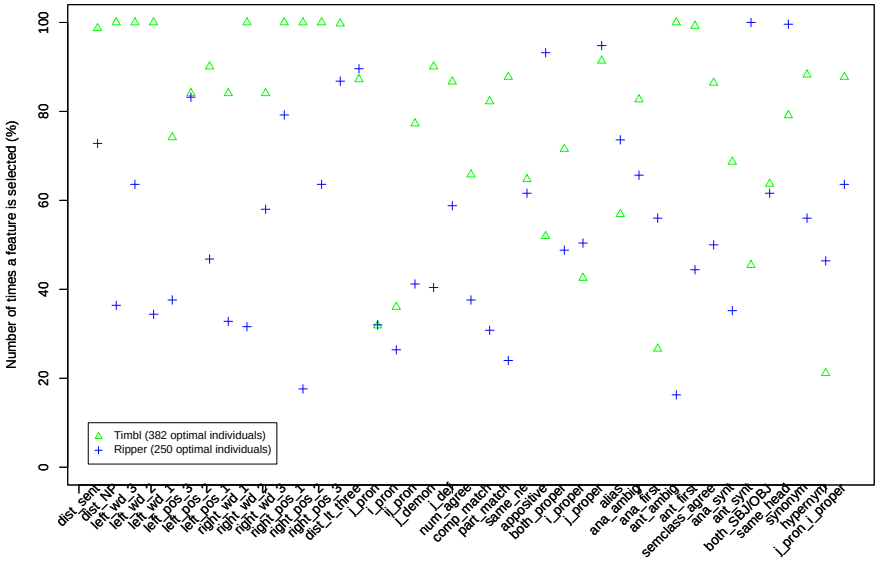


Figure 6.6: Number of times a feature is selected in the individuals for the "Common nouns" data.



exact match between the feature values is required. Furthermore, TIMBL also incorporates different feature weighting techniques to assign different degrees of informativeness to the selected features.

- With respect to the informativeness of the features, we can observe that all features are informative for our task of coreference resolution. This observation refines our earlier results displayed in Table 5.1 and also those reported by Soon et al. (2001), which show the lack of informativeness of the majority of the features, when they are considered in isolation. Furthermore, we can also observe that the initial predominance of the string-matching features (as also observed by Soon et al. (2001), Yang et al. (2004b) and others) has disappeared in favour of a more balanced combination of features.

- Furthermore, we can observe for all data sets that the feature selection considered to be optimal for TIMBL can be different from the one optimal for RIPPER. TIMBL and RIPPER often incorporate different features in their instances. An example: Figure 6.3 shows that RIPPER selects the "ant_first" feature (semantic class of the antecedent) 32% of the times, whereas this feature is part of 97% of the TIMBL individuals.

With respect to the selected parameters, we looked for general tendencies for each of both learning methods. Although the parameter settings which are selected after optimization cannot be generalized, not even within one single data set and although the parameter settings which are optimal when using all features are not necessarily optimal when performing feature selection, some general observations can be made.

For TIMBL, we can conclude the following. 99% of all optimal individuals consist of a combination of features for which the distance calculation is handled by the overlap metric and features handled by the MVDM metric. Furthermore, with respect to the different *feature weightings*, gain ratio is used as optimal weighting technique in 58% of the individuals. All others are selected in less than 16% of the cases. With respect to the *distance weights* we could observe that the different distance weighted class voting schemes, and especially inverse linear voting (61%), are preferred above the default majority voting (9%). Finally, considering the different selected values of k, the number of nearest distances taken into account, the default k=1 is only selected in 3% of the individuals. The largest part of the k values (51%) lies between k=10 and k=25 and 33% of the k values is between k=25 and k=50. This use of high k values can be explained trough the use of the MVDM metric in nearly all optimal individuals.

For RIPPER, the most noticeable observation concerns the *loss ratio* parameter, which allows to change the ratio of the cost of a false negative to the cost of a

false positive. In RIPPERs default settings, this parameter is set to 1. In all optimal individuals, however, this default setting is only selected in 3% of the cases. All other individuals have a loss ratio value below 1 (51% below 0.5 and 46% between 0.5 and 1) which implies that more importance is given to an improvement of the recall. In the default experiments, we could observe that RIPPER performed worse than TIMBL with respect to the recall scores. Optimization, and more specifically changing the loss ratio parameter leads to a focus on recall improvement. We will return to these results in the following chapter on learning from skewed data. With respect to the *class ordering* parameter, we could observe that the ordering method in which the classes are ordered by increasing frequency is selected in two thirds of the individuals (78%). The ordering method which orders the classes by decreasing frequency is never selected. This parameter selection choice can again be explained through the skewedness of the data. First, rules are learned for the positive minority class and the negative class is taken as default classification. The MDL option is chosen in 22% of the individuals. With respect to the number of *optimization passes* taken over the rules RIPPER learns, the default value 2 is selected in 91% of the individuals. And for the *hypothesis simplification* option, we could observe that the default 0.5 is only selected in 19% of the individuals. In 66% of the cases, more hypothesis simplification is preferred. Finally, no general conclusions could be drawn for the *negative tests* option (negative tests: 40%, no negative tests: 60%) and the *example coverage* option (0: 39%, 1: 18%, 2: 11%, 3: 32%).

6.4 Summary and discussion

The results reported in this and the previous chapters show that the variability recorded for the same algorithm when doing feature selection, algorithm parameter optimization and their joint optimization is often much larger than the difference between the two learning algorithms. These observations, however, are not limited to the task of anaphora resolution. In earlier work (Hoste et al. 2002, Daelemans and Hoste 2002, Daelemans, Hoste, De Meulder and Naudts 2003, Decadt et al. 2004), we came to similar conclusions for the task of word sense disambiguation, the prediction of diminutive suffixes and part-of-speech tagging. Table 6.2 clearly exemplifies the usefulness of parameter optimization for the task of word sense disambiguation. It shows the results on the English lexical sample data in the Senseval-3 competition.

Furthermore, this effect of optimization is not limited to natural language processing datasets. We performed experiments on 5 UCI benchmark datasets[2]: "database for fitting contact lenses" (24 instances), "contraceptive method choice"

[2]http://www/ics/uci/edu/~mlearn/MLRepository.html

	TRAINING SET		TEST SET			TRAINING SET		TEST SET	
LEMMA/POS	DEF	OPT	DEF	OPT	LEMMA/POS	DEF	OPT	DEF	OPT
provide/v	84.56	94.85	88.40	**92.75**	rule/n	75.44	91.23	50.00	**60.00**
eat/v	79.04	89.22	78.16	**91.95**	image/n	49.00	62.69	48.64	**56.75**
remain/v	85.40	95.62	82.85	**88.57**	paper/n	37.95	54.46	38.46	**55.55**
arm/n	88.67	93.20	84.21	**84.96**	produce/v	50.54	65.22	53.19	**55.31**
plan/v	67.93	78.48	75.00	**83.33**	suspend/v	46.34	59.35	34.37	**51.56**
add/v	73.95	82.38	79.54	**82.57**	argument/n	42.04	57.58	43.24	**51.35**
degree/n	64.56	78.38	71.09	**82.03**	difficulty/n	35.48	58.06	34.78	**39.13**
hot/a	68.67	78.00	76.74	**81.39**	performance/n	38.21	52.85	28.73	**39.08**
watch/v	85.71	89.80	78.43	**80.39**	use/v	80.77	88.46	**78.57**	**78.57**
smell/v	70.41	85.27	74.54	**78.18**	hear/v	64.52	74.19	**53.12**	**53.12**
bank/n	61.36	79.22	59.84	**78.03**	win/v	50.65	68.83	**48.71**	**48.71**
expect/v	64.93	77.92	73.07	**76.92**	different/a	54.81	65.27	**46.00**	**46.00**
talk/v	77.37	83.21	73.97	**75.34**	miss/v	40.00	68.89	**43.33**	**43.33**
appear/v	79.24	87.17	71.42	**75.18**	solid/a	9.80	31.78	**27.58**	**27.58**
decide/v	72.95	86.89	70.96	**74.19**	receive/v	75.00	80.77	**92.59**	88.88
wash/v	32.26	62.90	52.94	**73.52**	organization/n	67.66	77.51	69.64	**73.21**
mean/v	84.81	91.14	**77.50**	75.00	audience/n	73.90	85.29	**76.00**	74.00
party/n	61.82	71.96	65.51	**72.41**	operate/v	72.73	84.85	**66.66**	55.55
interest/n	63.28	70.36	59.13	**72.04**	write/v	64.29	71.43	**56.52**	43.47
express/v	48.62	72.48	45.45	**70.90**	play/v	48.42	64.21	**51.92**	42.30
sort/v	61.09	78.60	66.66	**70.83**	difference/n	57.14	68.51	**47.36**	46.49
treat/v	37.84	55.86	**40.35**	38.59	judgment/n	35.64	60.40	**40.62**	34.37
note/v	56.15	69.23	61.19	**68.65**	atmosphere/n	47.42	60.20	51.85	**70.37**
disc/n	54.03	69.19	52.00	**66.00**	encounter/v	51.94	65.89	58.46	**60.00**
climb/v	63.48	78.26	59.70	**64.17**	important/a	72.08	82.23	42.10	**47.36**
shelter/n	66.14	74.02	54.08	**63.26**	activate/v	70.40	80.27	64.91	80.70
simple/a	43.55	58.52	44.44	**61.11**	source/n	34.06	52.90	46.87	59.37
ask/v	49.80	62.06	60.30	**61.06**	OVERALL SCORE				
begin/v	53.41	63.07	53.16	**60.75**	FINE-GR.	59.82	71.28	60.80	**67.40**
lose/v	44.78	62.69	36.11	**52.77**	COARSE-GR.	/	/	/	**74.00**

Table 6.2: Classification accuracies for all lemmas in the English lexical sample task. The first column presents the words to be disambiguated, together with their part-of-speech. The second and third column present the default results and the optimized results of TIMBL on the validation data, whereas the last two columns contain the default and optimized scores on the official Senseval-3 test data.

(1473 instances), "breast-cancer-wisconsin" (699 instances), "car evaluation data base" (1728 instances) and "postoperative patient data" (90 instances). And we came to similar conclusions as on the NLP data sets, as shown in Table 6.3.

Table 6.3: Classification accuracies of TIMBL and RIPPER on 5 UCI benchmark datasets.

Dataset		TIMBL	RIPPER
Database for fitting contact lenses	Default	75.0	79.2
	GA optimization	87.5	87.5
Contraceptive method choice	Default	48.5	46.8
	GA optimization	54.8	49.8
Breast-cancer-wisconsin	Default	95.7	93.7
	GA optimization	97.6	95.7
Car evaluation database	Default	94.0	87.0
	GA optimization	96.9	98.4
Postoperative patient data	Default	55.6	71.1
	GA optimization	71.1	71.1

These effects explain why in the machine learning of natural language literature, so many results and interpretations about the superiority of one algorithm over the other are contradictory. The existing studies (e.g. Mooney (1996), Escudero et al. (2000), Ng and Lee (1996), Lee and Ng (2002) for the domain of word sense disambiguation) explore only a few points in the space of possible experiments for each algorithm to be compared. One example from this type of comparative work is the seminal paper from Mooney (1996) who tested seven machine learning algorithms with different biases on the ability to discover the different senses of the word "line". He concluded that within the class of symbolic machine learning methods, decision lists are at an advantage because of their rule ordering bias. Although the methodological set-up of the comparison in Mooney (1996) is sound and the results provide insight inside the role of algorithm bias, we showed that these results are not reliable since this comparative study (and also many others) is limited to default settings of algorithm parameters and a constant input representation. Through the optimization experiments, however, we showed that there is a high risk that other areas in the experimental space may lead to radically different results and conclusions. In general, the more effort is put in optimization, the more reliable the results and the comparison will be. We are well aware of the combinatorially explosive character of this type of optimization, but we believe that genetic algorithms are a computationally feasible way to achieve this.

In the following chapter, we add yet another dimension to this discussion on

factors influencing a machine learning experiment by investigating the effect of class distribution on classifier performance.

CHAPTER 7

The problem of imbalanced data sets

A general goal of classifier learning is to learn a model on the basis of training data which makes as few errors as possible when classifying previously unseen test data. Many factors can affect the success of a classifier: the specific 'bias' of the classifier, the selection and the size of the data set, the choice of algorithm parameters, the selection and representation of information sources and the possible interaction between all these factors. In the previous chapters, we experimentally showed for the eager learner RIPPER and the lazy learner TIMBL that the performance differences due to algorithm parameter optimization, feature selection, and the interaction between both easily overwhelm the performance differences between both algorithms in their default representation. We showed how we improved their performance by optimizing their algorithmic settings and by selecting the most informative information sources.

In this chapter, our focus shifts, away from the feature handling level and the algorithmic level, to the sample selection level. We investigate whether performance is hindered by the imbalanced class distribution in our data sets and we explore different strategies to cope with this skewedness. In Section 7.1, we introduce the problem of learning from imbalanced data. In the two following sections, we discuss different strategies for dealing with skewed class distributions. In Section 7.2, we discuss several proposals made in the machine learning

literature for dealing with skewed data. In Section 7.3, we narrow our scope to the problem of class imbalances when learning coreference resolution. In the remainder of the chapter, we focus on our experiments for handling the class imbalances in the MUC-6, MUC-7 and KNACK-2002 data sets.

7.1 Learning from imbalanced data sets

The problem of learning from data sets with an unbalanced class distribution occurs when the number of examples in one class is significantly greater than the number of examples in the other class. In other words, in an unbalanced data set the majority class is represented by a large portion of all the instances, whereas the other class, the minority class, has only a small part of all instances. For a multi-class classification task, it is also possible to have several minority classes.

One of the major reasons for studying the effect that class distribution can have on classifier learner, is that we are confronted with unbalanced data sets in many real-world applications. For all these applications it is crucial to know whether class imbalances affect learning and if so, how. Example applications include vision (Maloof 2003), credit card fraud detection (Chan and Stolfo 1998), the detection of oil spills in satellite radar images (Kubat, Holte and Matwin 1998) and language applications, such as text categorization (Lewis and Gale 1994), part-of-speech tagging, semantic class tagging and concept extraction (Cardie and Howe 1997). These studies and many others show that imbalanced data sets may result in poor performance of standard classification algorithms (e.g. decision tree learners, nearest neighbour and naive bayes methods). Some algorithms will find an acceptable trade-off between the false positive and true positive rates. Other algorithms often generate classifiers that maximize the overall classification accuracy, while completely ignoring the minority class.

The common approach in detection tasks such as credit card fraud detection, the detection of oil spills in satellite radar images, and NLP tasks such as text categorization and also coreference resolution is to define these tasks as **two-class classification** problems. This implies that the classifier labels instances as being "fraudulent" or "non-fraudulent" (credit card fraud detection), "oil spilling" or "non oil spilling" (oil spills in satellite radar images), "coreferential" or "non-coreferential" (coreference resolution), etc. But in all these tasks, we are only interested in the detection of fraud, oil spills or coreferential relations. From that perspective, we might consider these tasks as **one-class classification** (see for example Manevitz and Yousef (2001) and Tax (2001) for a discussion of one-class classification) problems.

The motivation to consider coreference resolution as a one-class classification task is that we are only given examples of one class, namely of coreferential relations between NPs and we wish to determine whether a pair of NPs is coreferential. But the negative "non-coreferential" class can be anything else, which makes the choice of negative data for this task arbitrary, as shown in Section 7.3. The number of possible candidates for building negative instances is so huge, that finding interesting instances, or instances near the positive instances, is challenging. To train a standard two-class classification algorithm will probably result in a high number of false negative detections. However, considering the coreference resolution task as a one-class classification task requires the use of an entirely different classification strategy (such as one-class support vector machines (Tax 2001)) as the one being used in this thesis.

Since the difference between one-class and two-class classification is beyond the scope of this work, we will restrict the discussion to the task of coreference resolution as a two-class classification task. The positive class ("coreferential") will always correspond to the minority class and the negative class ("non-coreferential") to the majority class.

7.2 Machine learning research on imbalanced data sets

A central question in the discussion on data sets with an imbalanced class distribution is in what proportion the classes should be represented in the training data. One can argue that the natural class distribution should be used for training, even if it is highly imbalanced, since a model can then be built which fits a similar imbalanced class distribution in the test set. Others believe that the training set should contain an increased number of minority class examples. In the machine learning literature, there have been several proposals (see Japkowicz and Stephen (2002)) for adjusting the number of majority class and minority class examples. Methods include resizing training data sets or sampling, adjusting misclassification costs, learning from the minority class, adjusting the weights of the examples, etc. We will now discuss these approaches in more detail. In Subsection 7.2.1, we discuss two commonly used methods to adapt machine learning algorithms to imbalanced classes: under-sampling and over-sampling. We continue with a discussion on cost-sensitive classifiers. Subsection 7.2.3 covers the approaches in which the examples are weighted in an effort to bias the performance to the minority class.

7.2.1 Sampling

Two sampling methods are commonly used to adapt machine learning algorithms to imbalanced classes: **under-sampling or down-sampling** and **over-sampling or up-sampling**. In case of under-sampling, examples from the majority class are removed. Examples removed can be randomly selected, or near miss examples, or examples that are far from the minority class examples. In case of over-sampling, examples from the minority class are duplicated. Both sampling techniques can also be combined. Examples of this type of sampling research include Kubat, Holte and Matwin (1997), Chawla, Bowyer, Hall and Kegelmeyer (2002), Drummond and Holte (2003) and Zhang and Mani (2003). The primary motivation of the use of sampling for skewed data sets is to improve classifier performance. But under-sampling can also be used as a means to reduce training set size.

Especially the sensitivity of the C4.5 decision tree learner to skewed data sets and the effect of under-sampling and over-sampling on its performance has been intensively studied. Drummond and Holte (2003), Domingos (1999), Weiss (2003), Japkowicz and Stephen (2002) and Joshi, Kumar and Agarwal (2001) all investigate the effect of class distribution on the C4.5 classifier. The conclusions are similar[1]: under-sampling leads to better results, whereas over-sampling produces little or no change in performance. None of the approaches, however, consistently outperforms the other and it is also difficult to determine a specific under-sampling or over-sampling rate which consistently leads to the best results. We will come to similar conclusions for our experiments.

Both over-sampling and under-sampling have known **drawbacks**. The major drawback from under-sampling is that it disregards possibly useful information. This can be countered by more intelligent under-sampling strategies such as those proposed by Kubat et al. (1997) and Chan and Stolfo (1998). Kubat et al. (1997), for example, consider majority examples which are close to the minority class examples as noise and discard these examples. Chan and Stolfo (1998) choose for an under-sampling approach without any loss of information. In a preliminary experiment, they determine the best class distribution for learning and then generate different data sets with this class distribution. This is accomplished by randomly dividing the majority class instances. Each of these data sets then contains all minority class instances and one part of the majority class instances. The sum of the majority class examples in all these data sets is the complete set of majority class examples in the training set. They then learn a classifier on these different data sets and integrate all these classifiers (meta-learning) by learning from their classification behaviour.

[1] Except for Japkowicz and Stephen (2002) who come to the opposite conclusion.

One of the problems with over-sampling is that it increases the size of the training set and the time to build a classifier. Furthermore, in case of decision tree learning, the decision region for the minority class becomes very specific through the replication of the minority class and this causes new splits in the decision tree, which can lead to overfitting. It is possible that classification rules are induced which cover one single copied minority class example. An over-sampling strategy which aims to make the decision region of the minority class more general and hence aims to counter overfitting has been proposed by Chawla et al. (2002). They form new minority class examples by interpolating between minority class examples that lie close together.

Although most of the sampling research focuses on decision tree learning, this does not imply that other learning techniques are immune to the class distribution of the training data. Also support vector machines (Raskutti and Kowalczyk 2003), kNN methods (Zhang and Mani 2003) (see also 7.2.3), neural networks (Zhang, Mani, Lawrence, Burns, Back, Tsoi and Giles 1998), etc. have been shown to be sensitive to the class imbalances in the data set.

7.2.2 Adjusting misclassification costs

Another approach for coping with skewed data sets is the use of cost-sensitive classifiers. If we consider the following cost matrix, it is obvious that the main objective of a classifier is to minimize the false positive and false negative rates.

	Actual negative	Actual positive
Predict negative	true negative	false negative
Predict positive	false positive	true positive

If the number of negative and positive instances is highly unbalanced, this will typically lead to a classifier which has a low error rate for the majority class and a high error rate for the minority class. Cost-sensitive classifiers (Pazzani, Merz, Murphy, Ali, Hume and Brunk 1994, Domingos 1999, Kubat et al. 1998, Fan, Stolfo, Zhang and Chan 1999, Ting 2000, Joshi et al. 2001) have been developed to handle this problem by trying to reduce the cost of misclassified examples, instead of classification error. Cost-sensitive classifiers may be used for unbalanced data sets by setting a high cost to the misclassifications of a minority class example.

The MetaCost algorithm of Domingos (1999) is an example of such a cost-sensitive classifier approach. It uses a variant of bagging (Breiman 1996), which

makes bootstrap replicates from the training set by taking samples with replacement from the training set. In MetaCost, multiple bootstrap samples are made from the training set and classifiers are trained on each of these samples. The class's probability for each example is estimated by the fraction of votes that it receives from the ensemble. The training examples are then relabeled with the estimated optimal class and a classifier is reapplied to this relabeled data set. Domingos (1999) compared his approach with under-sampling and over-sampling and showed that the MetaCost approach is superior to both.

Other cost-sensitive algorithms are the boosting [2] algorithms CSB1, CSB2 (Ting 2000) and AdaCost (Fan et al. 1999). In order to better handle data sets with rare cases, these algorithms take into account different costs of making false positive predictions versus false negative predictions. So in contrast to AdaBoost (Freund and Schapire 1996), in which a same weight is given to false and true positives and false and true negatives, the CSB1, CSB2 and AdaCost algorithms update the weights of all four types of examples differently. All three algorithms assign higher weights to the false negatives and thus focus on a recall improvement.

7.2.3 Weighting of examples

The 'weighting of examples' approach has been proposed from within the case-based learning framework (Cardie and Howe 1997, Howe and Cardie 1997). It involves the creation of specific weight vectors in order to improve minority class predictions. The commonly used feature weighting approach is the use of so-called task-based feature weights (as for example also used in TIMBL), in which the feature weights are calculated for the whole instance base.

In order to increase the performance on the minority class, however, Cardie and Howe (1997) and Howe and Cardie (1997) propose the use of class-specific and even test-case-specific weights. The class-specific weights are calculated per class whereas the test-case-specific weights are calculated for each single instance. The creation of class-specific weights (Howe and Cardie 1997) is as follows: the weights for a particular class on a given feature are based on the distribution of feature values for the instances in that class and the distribution of feature values for the instances in the other class(es). Highly dissimilar distributions imply that the feature can be considered useful and will have a high weight.

[2] Boosting is a machine learning method in which learning starts with a base learning algorithm (e.g. C4.5 (Quinlan 1996)), which is invoked many times. Initially, all weights over the original training set are set equally. But on each boosting round, these weights are adjusted: the weights of incorrectly classified examples are increased, whereas the weights of the correctly classified examples are decreased. Through these weight adjustments, the classifier is forced to focus on the hard training examples.

During testing, all training instances with the same class value are assigned the weight associated with that particular class value. Howe and Cardie (1997) describe different techniques for the creation of class-specific weights. These techniques represent different levels of locality in feature weighting, ranging from the calculation of feature weight vectors across all classes to get a single global weight vector to a fine-grained locality by assigning different weights for each individual feature value. They show that the use of class-specific weights globally leads to better classification accuracy.

Cardie and Howe (1997) describe the use of test-case-specific weights, which are determined on the basis of decision trees. The weight vector for a given test case is calculated as follows: (1) Present the test case to the decision tree and note the path that is taken through the tree, (2) Omit the features that do not appear along this path, (3) calculate the weights for the features that appear along the path by using path-specific information gain values, (4) use this weight vector in the learning algorithm to determine the class of the test case. Cardie and Howe (1997) show that example weighting leads to a significant increase of the recall.

In the experiments described in the remainder of this chapter in which we investigate the effect of class distribution on classifier performance, we decided not to introduce additional learning techniques, such as decision tree learning or different boosting techniques in this discussion on methodology. Instead, we chose for a straight-forward resampling procedure and a variation of the internal loss ratio parameter in RIPPER.

7.3 Imbalanced data sets in coreference resolution

As already frequently mentioned before, coreference resolution data sets reveal large class imbalances: only a small part of the possible relations between noun phrases is coreferential (see for example Table 3.1). When trained on such imbalanced data sets, classifiers can exhibit a good performance on the majority class instances but a high error rate on the minority class instances. Always assigning the "non coreferential" class will lead to a highly 'accurate' classifier, which cannot find any coreferential chain in a text.

7.3.1 Instance selection in the machine learning of coreference resolution literature

In the machine learning of coreference resolution literature, this problem of class imbalances has to our knowledge not yet been thoroughly investigated. However, the different methodologies for corpus construction show that at least the problem of instance selection has been acknowledged. Soon et al. (2001), for example, only create positive training instances between anaphors and their immediately preceding antecedent. The NPs occurring between the two members of each antecedent-anaphor pair are used for the creation of the negative training examples. Imposing these restrictions on corpus construction still leads to high imbalances: in their MUC-6 and MUC-7 training data, only 6.5% and 4.4%, respectively, of the instances is positive. Strube et al. (2002) use the same methodology as Soon et al. (2001) for the creation of positive and negative instances, but they also first apply a number of filters, which reduce up to 50% of the negative instances. These filters are all linguistically motivated, e.g. discard an antecedent-anaphor pair (i) if the anaphor is an indefinite NP, (ii) if one entity is embedded into the other, e.g. if the potential anaphor is the head of the potential antecedent NP, (iii) if either pronominal entity has a value other than third person singular or plural in its agreement feature. But Strube et al. (2002) do not report results of experiments before and after application of these linguistic filters. And Yang et al. (2003) use the following filtering algorithm to reduce the number of instances in the training set: (i) add the NPs in the current and previous two sentences and remove the NPs that disagree in number, gender and person in case of pronominal anaphors, (ii) add all the non-pronominal antecedents to the initial candidate set in case of non-pronominal anaphors. But also here, no comparative results are provided of experiments with and without instance selection.

Ng and Cardie (2002a) propose both negative sample selection (the reduction of the number of negative instances) and positive sample selection (the reduction of the number of positive instances), both under-sampling strategies aiming to create a better coreference resolution system. Ng and Cardie (2002a) use a technique for negative instance selection, similar to that proposed in Soon et al. (2001) and they create negative instances for the NPs occurring between an anaphor and its farthest antecedent. Furthermore, they try to avoid the inclusion of hard training instances. Given the observation that one antecedent is sufficient to resolve an anaphor, they present a corpus-based method for the selection of easy positive instances, which is inspired by the example selection algorithm introduced in Harabagiu et al. (2001). The assumption is that the easiest types of coreference relationships to resolve are the ones that occur with high frequencies in the training data. Harabagiu et al. (2001) mine by hand three sets of coreference rules for covering positive instances from the training data by

finding the coreference knowledge satisfied by the largest number of anaphor-antecedent pairs. The high confidence coreference rules, for example, look for (i) repetitions of the same expression, (ii) appositions or arguments of the same copulative verb, (iii) name alias recognitions, (iv) anaphors and antecedents having the same head. Whenever the conditions for a rule are satisfied, an antecedent for the anaphor is identified and all other pairs involving the same anaphor can be filtered out. Ng and Cardie (2002a) write an automatic positive sample selection algorithm that coarsely mimics the Harabagiu et al. (2001) algorithm by finding a confident antecedent for each anaphor. They show that system performance improves dramatically with positive sample selection. The application of both negative and positive sample selection leads to even better performance. But they mention a drawback in case of negative sample selection: it improves recall but damages precision.

All these approaches concentrate on instance selection, on a reduction of the training material and they aim to produce better performing classifiers through the application of linguistically motivated filters on the training data before application of the classifier. Through the application of these linguistic filters, part of the problem to be solved, viz. coreference resolution, is solved beforehand.

Our instance selection approach differs from these approaches on the following points:

- We investigate whether both learning approaches we experiment with are sensitive to class imbalances in the training data. None of the above described approaches investigates the effect of class imbalances on classifier performance.
- In case of sensitivity to class imbalances, we investigate whether classifier performance can be improved through a rebalancing of the data set. This rebalancing is done without any a priori linguistic knowledge about the task to be solved.

7.3.2 Investigating the effect of skewedness on classifier performance

In Section 3.1.2, we described the selection of positive and negative instances for the training data. For the construction of these instances, we did not impose any limitations on the construction of the instance base. For English, we did not take into account any restrictions with respect to the maximum distance between a given anaphor and its antecedent. Due to the presence of documents exceeding 100 sentences in the KNACK-2002 data, negative instances were only

made for the NPs in a range of 20 sentences preceding the candidate anaphor. For both languages, we did not apply any linguistic filters (such as gender and number agreement between both nominal constituents) on the construction of the positive and negative instances. The main "restriction" was that, since we investigate anaphoric and not cataphoric relations, we only looked back in the text for the construction of the instances. The instances were made as follows:

- **Positive instances** were made by combining each anaphor with each preceding element in the coreference chain.
- The **negative instances** were built (i) by combining each anaphor with each preceding NP which was not part of any coreference chain and (ii) by combining each anaphor with each preceding NP which was part of another coreference chain.

Table 7.1: $F_{\beta=1}$ and recall results on the cross-validation data in relation to the share of minority class examples in the data sets.

		$F_{\beta=1}$		Recall	
MUC-6	% of min. class inst.	Timbl	Ripper	Timbl	Ripper
All	6.6	56.15	**62.59**	**55.50**	49.65
Pronouns	7.0	**31.97**	28.70	**27.42**	19.44
Proper nouns	7.9	65.37	**71.04**	**67.53**	61.60
Common nouns	5.0	53.62	**65.44**	53.53	**55.55**
MUC-7					
All	5.80	48.68	**49.36**	**46.09**	36.21
Pronouns	8.54	**39.25**	32.86	**36.60**	22.70
Proper nouns	6.00	59.49	**64.83**	**56.87**	52.56
Common nouns	4.24	41.03	**49.24**	**39.17**	36.76
KNACK-2002					
All	6.31	**46.78**	46.49	**44.93**	34.92
Pronouns	8.58	47.31	**50.57**	**44.81**	43.14
Proper nouns	6.18	58.11	**60.21**	**54.04**	49.49
Common nouns	3.92	30.51	**36.52**	**30.37**	25.92

As shown in Table 7.1, this approach leads to an instance base with a highly skewed class distribution for all three data sets. In the MUC-6 training data, for example, 159,815 instances out of 171,081 are negative and merely 11,266 (6.6% of the total) are positives. Furthermore, the number of instances in both training and test set is large comparing to the number of references (in MUC-6 1644 and

1627 respectively) present in both sets. In the KNACK-2002 cross-validation data, for example, merely 6.3% of the instances is classified as positive. But is learning performance hindered when learning from these data sets where the minority class is underrepresented?

Table 7.1 shows that although RIPPER performs better on the data set as a whole, it exhibits a poorer performance on the minority class than TIMBL does. The $F_{\beta=1}$ results in Table 7.1 show that RIPPER outperforms TIMBL in 9 out of 12 results. But with respect to the recall scores, which is the number of correctly classified minority class examples, the opposite tendency can be observed: TIMBL generally (11 out of 12 results) obtains a higher recall than RIPPER, which implies that TIMBL produces fewer false negatives. We believe that this can be explained by the nature of both learning approaches. In a lazy learning approach, all examples are stored in memory and no attempt is made to simplify the model by eliminating low frequency events. In an eager learning approach such as RIPPER, however, possibly interesting information from the training data is either thrown away by pruning or made inaccessible by the eager construction of the model. This type of approach abstracts from low-frequency events. Applied to our data sets, this implies that RIPPER will prune away possibly interesting low-frequency positive data. We will return to this issue in Section 7.4.

In the previous section, we discussed several techniques proposed in the machine learning literature for handling data sets with skewed class distributions, including up-sampling, down-sampling, adjusting misclassification costs, etc. In the following section, we will investigate some of these techniques and will evaluate the effect of class distribution and training set size on the performance of TIMBL and RIPPER.

7.4 Balancing the data set

In order to investigate the effect of class distribution on classifier performance, it is necessary to compare the performance of the classifier on training sets with a variety of class distributions. One possible approach to create this variety of class distributions is to decrease the number of instances in the majority class. We investigated the effect of random down-sampling and down-sampling of the true negatives for both TIMBL and RIPPER. For RIPPER, we also changed the ratio false negatives and false positives in order to improve recall. We did not perform any up-sampling experiments, since creating multiple copies from one instance can only guide the choice of classification in memory-based learning if there is a conflict among nearest neighbours. Furthermore, as already discussed

earlier, up-sampling can lead to rules overfitting the training data. For example, when a certain instance is copied ten times, the rule learner might quite possibly form a rule to cover that one instance.

7.4.1 Random

In order to reduce the number of negative training instances, we experimented with two randomized down-sampling techniques. In a first experiment, we gradually down-sampled the majority class at random. We started off with no down-sampling at all and then gradually downsized the number of negative instances in slices of 10% until there was an equal number of positive and negative training instances.

With respect to the accuracy results, we can conclude for both classifiers that the overall classification accuracy decreases with a decreasing rate of negative instances. The precision, recall and $F_{\beta=1}$ results from the down-sampling experiments for both TIMBL and RIPPER are plotted in Figure 7.1, Figure 7.2 and Figure 7.3. At the X-axis, the different down-sampling levels are listed, ranging from 1 (no down-sampling at all) to 0 (an equal number of positive and negative instances). The plots for both learning methods show that a decreasing rate of negative instances is beneficial for the recall or the classification accuracy on the minority class instances. The plots also reveal that down-sampling is harmful for precision. By reducing the number of negative instances, it becomes more likely for a test instance to be classified as positive. This implies that more negative instances will be classified as being positive (the false positives). But the plots also reveal more subtle tendencies.

For TIMBL, the $F_{\beta=1}$ values for the "Pronouns" data set in the whole down-sampling process remain rather constant (i.e. there are no significant differences compared to the default scores) or they even significantly increase. On MUC-6, for example, TIMBL obtains a default $F_{\beta=1}$ score of 31.97% and decreasing the number of negative instances with 40% leads to a top $F_{\beta=1}$ score of 34.42%. For MUC-7, TIMBL obtains $F_{\beta=1}$ scores ranging between 38.21% and 41.41%. Only the last down-sampling step in which the training set contains an equal number of positive and negative instances leads to a larger drop in $F_{\beta=1}$ scores (34.21%). For KNACK-2002, the default $F_{\beta=1}$ score of 47.31% is raised to 50.02% at a down-sampling level of 0.5. For the other three data sets ("All", "Proper nouns" and "Common nouns"), however, down-sampling does not lead to a significant increase of $F_{\beta=1}$ values.

With respect to the RIPPER $F_{\beta=1}$ values, we can conclude the following. For MUC-6, the $F_{\beta=1}$ values for the "Pronouns" data set are all significantly better

than the default result during the whole down-sampling process. RIPPER obtains a default $F_{\beta=1}$ score of 28.70% and decreasing the number of negative instances with 90% leads to a top $F_{\beta=1}$ score of 37.04%. A similar tendency can be observed for MUC-7: a default $F_{\beta=1}$ score of 32.86% and a top $F_{\beta=1}$ score of 45.00% when down-sampling with 80%. For KNACK-2002, the default $F_{\beta=1}$ score of 50.57% is raised to 63.00% at a down-sampling level of 0.5. For the other three data sets, the $F_{\beta=1}$ values only significantly deteriorate at higher down-sampling levels. For MUC-6, for example, the $F_{\beta=1}$ values for the "Proper nouns", "Common nouns" and "All" data sets only significantly deteriorate when down-sampling with 30%, 70% and 50% respectively. For the KNACK-2002 data, down-sampling even leads to a significant improvement on the default $F_{\beta=1}$ score: from a default 46.49% to a top score 58.84% at level 0.3 ("All"), from 36.52% to 42.84% at level 0.3 ("Common nouns") and from 60.21% to 64.60% at level 0.3 ("Proper nouns").

For all these random down-sampling experiments we can conclude that TIMBL and RIPPER behave differently. In Table 7.1, we showed that RIPPER is more sensitive to the skewedness of the classes. A comparison of results in the Figures 7.1, 7.2 and 7.3 shows that down-sampling can be beneficial for the RIPPER results. Furthermore, down-sampling only starts being harmful at a high down-sampling level. TIMBL has shown this tendency only on the "Pronouns" data set. All these observed tendencies hold for both English data sets and the Dutch data set.

7.4.2 Exploiting the confusion matrix

In the previous sampling experiments, all negative instances (both the true negatives and false positives) qualified for down-sampling. In a new set of experiments, we also experimented with down-sampling of the true negatives. This implies that the falsely classified negative instances, the false positives, were kept in the training set, since they were considered harder to classify. The true negatives were determined in a leave-one-out experiment on the different training folds and then down-sampled. However, due to the highly skewed class distribution, the level of true negatives is very high, e.g. Timbl falsely classifies merely 3% of the negative instances in the KNACK-2002 "All" data set. This implies that these down-sampling experiments reveal the same tendencies as the random down-sampling experiments.

As already touched upon in Section 4.3, RIPPER incorporates a loss ratio (Lewis and Gale 1994) (see also Section 4.3) parameter which allows the user to specify the relative cost of two types of errors: false positives and false negatives. It thus controls the relative weight of precision versus recall. In its default version,

Figure 7.1: Cross-validation results in terms of precision, recall and $F_{\beta=1}$ after application of TIMBL and RIPPER on the MUC-6 data with a randomly downsampled majority class. The test partitions keep their initial class distribution.

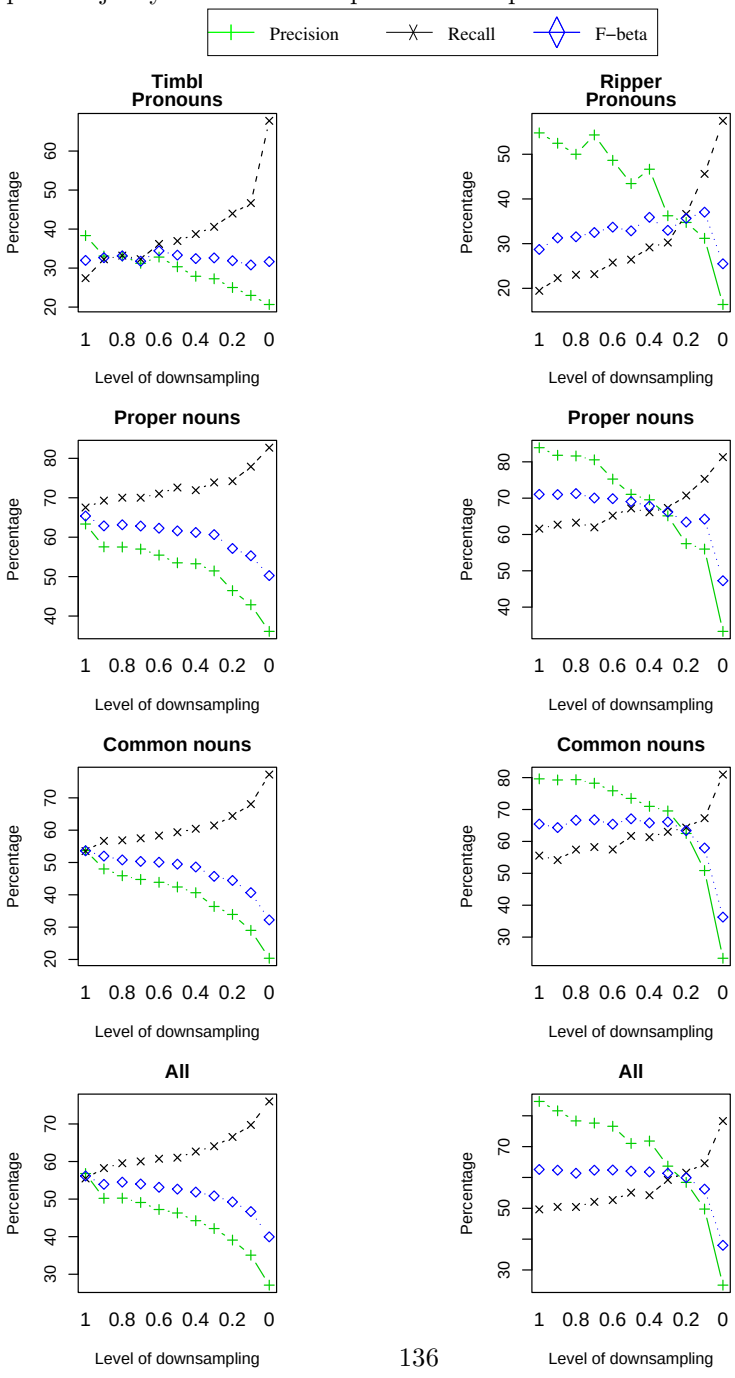


Figure 7.2: Cross-validation results in terms of precision, recall and $F_{\beta=1}$ after application of TIMBL and RIPPER on the MUC-7 data with a randomly down-sampled majority class. The test partitions keep their initial class distribution.

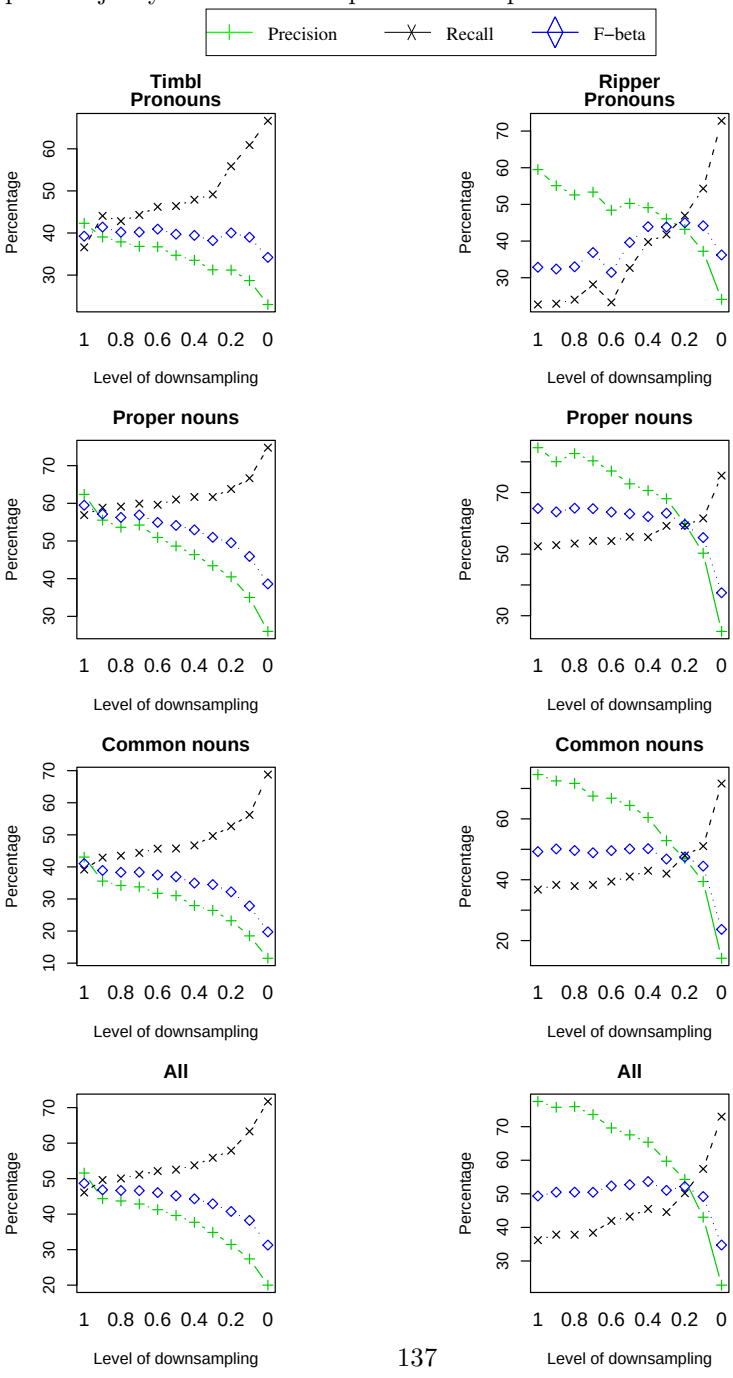
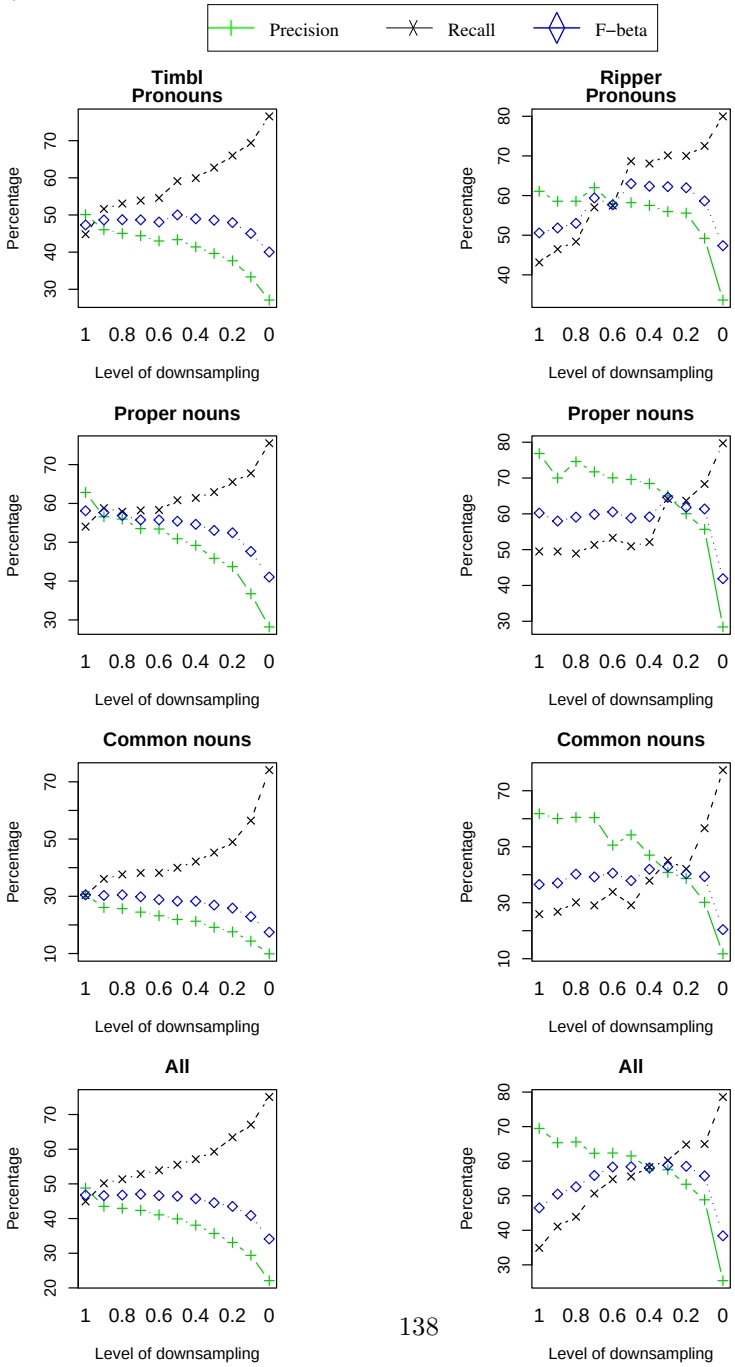


Figure 7.3: Cross-validation results in terms of precision, recall and $F_{\beta=1}$ after application of TIMBL and RIPPER on the KNACK-2002 data with a randomly down-sampled majority class. The test partitions keep their initial class distribution.



RIPPER uses a loss ratio of 1, which indicates that the two errors have equal costs. A loss ratio greater than 1 indicates that false positive errors (where a negative instance is classified positive) are more costly than false negative errors (where a positive instance is classified negative). Setting the loss ratio above 1 can be used in combination with the up-sampling of the positive minority class in order to counterbalance the overrepresentation of the positive instances. But this is not what we need. In all previous experiments with RIPPER we could observe high precision scores and rather low recall scores. Therefore, we decided to focus the loss ratio on the recall. For our experiments, we varied the loss ratio in RIPPER from 1 (default) to 0.05 (see for example also Chawla et al. (2002) for similar experiments). The motivation for this reduction of the loss ratio is double: (i) improve on recall and (ii) build a less restrictive set of rules for the minority class.

We can conclude from these experiments that, as also observed in the down-sampling experiments, a change of loss ratio is generally at the cost of overall classification accuracy. The cross-validation precision, recall and $F_{\beta=1}$ results of these experiments are displayed in Figure 7.4, Figure 7.5 and Figure 7.6. Similar tendencies as for the down-sampling experiments can be observed: the focus on recall is harmful for precision. With respect to the $F_{\beta=1}$ values, we can conclude that the $F_{\beta=1}$ values for the "Pronouns" can significantly increase when decreasing the loss ratio value. On MUC-6, for example, RIPPER obtains a default $F_{\beta=1}$ score of 28.70% and decreasing the loss ratio value to 0.09 leads to a top $F_{\beta=1}$ score of 38.80%. On MUC-7, the $F_{\beta=1}$ score is raised up to 13% when changing the loss ratio parameter. With respect to the MUC-6 and MUC-7 $F_{\beta=1}$ values for the "Proper nouns", "Common nouns" and "All" data sets we can observe a small increase of performance. As shown in Figure 7.6 and Table 7.2, a change of the loss ratio parameter leads to a large performance increase for the different KNACK-2002 data sets. Furthermore, for three out of the four data sets, the default class distribution returns the lowest $F_{\beta=1}$ score.

Table 7.2: Default $F_{\beta=1}$ results for the KNACK-2002 data, in comparison with the highest and lowest scores after change of the loss ratio parameter.

	default	high	low
All	46.49	60.33 (loss ratio: 0.2)	46.49
Pronouns	50.57	63.49 (loss ratio: 0.4)	50.57
Proper nouns	60.21	63.69 (loss ratio: 0.06)	58.61
Common nouns	36.52	42.68 (loss ratio: 0.07)	36.52

The general conclusion from these experiments with loss ratio reduction is that decreasing the loss ratio leads to better recall at the cost of precision. For both

the English and Dutch data sets overall $F_{\beta=1}$ increases can be observed. Furthermore, loss ratio reduction also leads to a less restrictive set of rules for the minority class, reflected in an increasing recall. With respect to the specific loss ratio values, we conclude that no particular value leads to the best performance over all data sets. This confirms our findings in the parameter optimization experiments (Chapter 5), which also revealed that the optimal parameter settings of an algorithm for a given task have to be determined experimentally for each new data set.

7.5 Summary and discussion

In this chapter we focused on the problem of imbalanced data sets. In Section 7.2, we discussed several proposals made in the machine learning literature for dealing with skewed data sets and we continued with a discussion on class imbalances when learning coreference resolution. In the remainder of the chapter, we presented results for the MUC-6, MUC-7 and KNACK-2002 data sets. We first compared the share of minority class examples in the data sets with the percentage of the total test errors that can be attributed to misclassified minority class test examples. There, we could observe a large number of false negatives or a large error rate for the examples from the minority class. These results confirm earlier results on both machine learning and NLP data sets, e.g. by Weiss (2003) or Cardie and Howe (1997). Furthermore, we showed that although RIPPER performs better on the data set as a whole, it exhibits a poorer performance on the recall for the minority class than TIMBL does.

In order to investigate the effect of class distribution on the performance of TIMBL and RIPPER, we created a variety of class distributions through the use of down-sampling and by changing the loss ratio parameter in RIPPER. For the down-sampling experiments we could conclude for the two learning methods that a decreasing rate of negative instances is beneficial for recall. The same conclusion could be drawn in the experiments in wich the loss ratio parameter was varied for RIPPER. These conclusions confirm earlier findings of Chan and Stolfo (1998), Weiss (2003) and others. Another general conclusion is that both down-sampling and a change of the loss ratio parameter below 1 is harmful for precision. This implies that more false positives will be produced.

However, both learning approaches behave quite differently in case of skewedness of the classes and they also react differently to a change in class distribution. TIMBL, which performs better on the minority class than RIPPER in case of a largely imbalanced class distribution, mainly suffers from a rebalancing of the data set. In contrast, the RIPPER results are sensitive to a change of class

Figure 7.4: Cross-validation results in terms of precision, recall and $F_{\beta=1}$ when changing the RIPPER loss ratio parameter on the MUC-6 data.

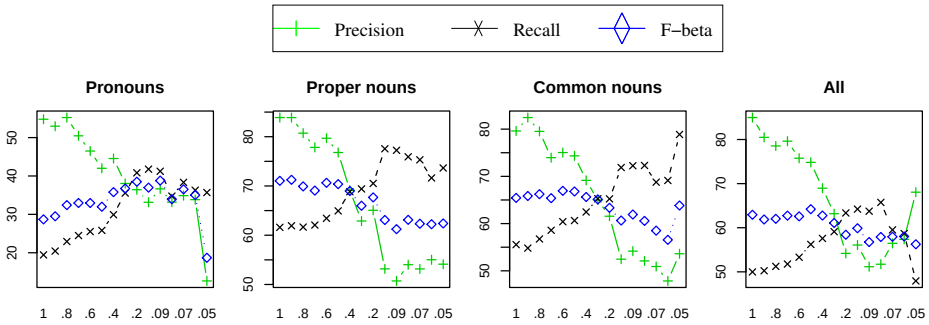


Figure 7.5: Cross-validation results in terms of precision, recall and $F_{\beta=1}$ when changing the RIPPER loss ratio parameter on the MUC-7 data.

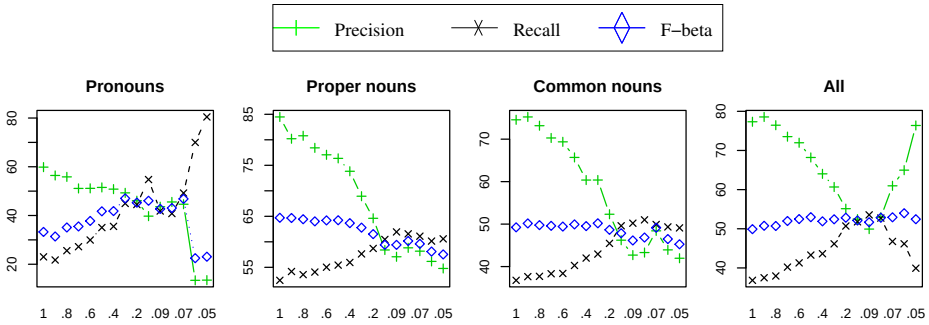
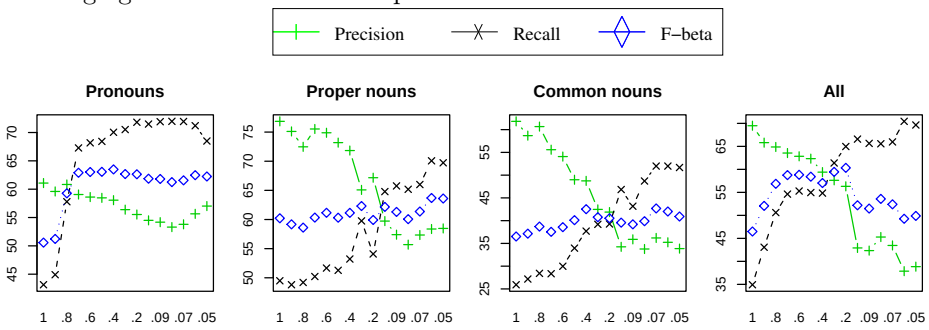


Figure 7.6: Cross-validation results in terms of precision, recall and $F_{\beta=1}$ when changing the RIPPER loss ratio parameter on the KNACK-2002 data.



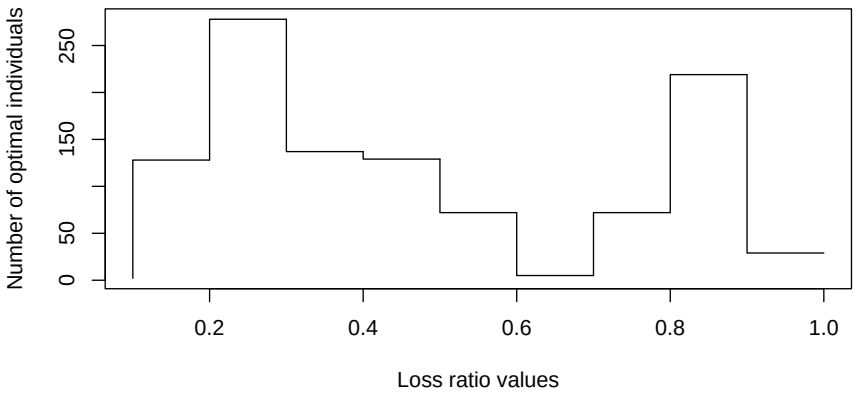
distribution or loss ratio. We believe that this different behaviour of TIMBL and RIPPER can be explained by the nature of both learning approaches (see also Daelemans et al. (1999) for a discussion on this topic). In a lazy learning approach, all examples are stored in memory and no attempt is made to simplify the model by eliminating low frequency events. In an eager learning approach such as RIPPER, however, possibly interesting information from the training data is either thrown away by pruning or made inaccessible by the eager construction of the model. This type of approach makes abstraction from low-frequency events. Applied to our data sets, this implies that RIPPER prunes away possibly interesting low-frequency positive data. A decrease of the number of negative instances counters this pruning.

This chapter also concludes our discussion on different factors which can influence a (comparative) machine learning experiment. Throughout the previous chapters we experimentally showed that apart from algorithm bias, many other factors such as data set selection, the selection of information sources and algorithm parameters and also their interaction potentially play a role in the outcome of a machine learning experiment. We showed that changing any of the architectural variables can have great effects of the performance of a learning method, making questionable many conclusions in the literature based on default settings of algorithms or on partial optimization only.

In the algorithm-comparing experiments using all information sources and default parameter settings, we could observe some clear tendencies with respect to the precision and recall scores. We saw that the precision scores for TIMBL were up to about 30% lower than the ones for RIPPER, which implies that TIMBL falsely classifies more instances as being coreferential. Furthermore, with respect to the recall scores, the opposite tendency could be observed: TIMBL generally obtained a higher recall than RIPPER. In the feature selection experiments, we observed the large effect feature selection can have on classifier performance. Especially TIMBL showed a big sensitivity to a good feature subset. In the parameter optimization experiments we observed that the performance differences within one learning method are much larger than the method-comparing performance differences, which was also confirmed in the experiments exploring the interaction between feature selection and parameter optimization. Furthermore, with respect to the selected features and parameter settings, we observed that no particular parameter setting and no particular feature selection is optimal. This implies that the parameter settings which are optimal using all features are not necessarily optimal when performing feature selection. We also showed that the features considered to be optimal for TIMBL can be different than the ones optimal for RIPPER. In the experiments varying the class distribution of the training data, we showed that this was primarily beneficial for RIPPER. Once again, we showed that no particular class distribution nor loss ratio value was

optimal for all data sets. Therefore, this resampling should also be subject to optimization. This additional optimization step was already incorporated in the joint feature selection and parameter optimization experiments reported in the previous chapter, where we also varied the loss ratio parameter for RIPPER. These experiments revealed that a loss ratio value below one was selected in 97% of the optimal individuals found over all experiments. An illustration of these selected loss ratio values is given in Figure 7.7.

Figure 7.7: Selected loss ratio values in the individuals found to be optimal after GA optimization.



CHAPTER 8

Testing

In all previous chapters, we reported cross-validation results on the training data. Defining the anaphora resolution process as a classification problem, however, involves the use of a two-step procedure. In a **first step**, the classifier (in our case TIMBL or RIPPER) decides on the basis of the information learned from the training set whether the combination of a given anaphor and its candidate antecedent in the test set is classified as a coreferential link. Since each NP in the test set is linked with several preceding NPs, this implies that one single anaphor can be linked to more than one antecedent, which for its part can also refer to multiple antecedents, and so on. Therefore, a **second step** is taken, which involves the selection of one coreferential link per anaphor.

In the previous chapters, we focused on the first step by trying to reach the optimal result through feature selection, algorithm parameter optimization and different sampling techniques. In this chapter, we move away from the instance level and concentrate on the coreferential chains. This requires a new experimental setup (Section 8.1) with a new evaluation procedure (Section 8.2). In Section 8.3, we report the results of TIMBL and RIPPER on the different data sets. Section 8.4 describes the main observations from a qualitative error analysis on a selection of English and Dutch documents.

8.1 Data preparation

The general setup of our experiments on the test set is the following. For all three data sets (MUC-6, MUC-7 and KNACK-2002), we use a held-out test set. Both RIPPER and TIMBL are trained on the complete training set and the resulting classifiers are applied to the held-out test set, which is represented as a set of instances. For a description of the different preprocessing steps and the construction of the features we refer to Subsection 3.1.1 and Section 3.2. The construction of the test instances, however, is different from that for the training instances. For the construction of these test instances, all NPs starting from the second NP in a text are considered a possible anaphor, whereas all preceding NPs are considered possible antecedents of the anaphor under consideration. Since this type of instance construction leads to an enormous increase of the data set and since we are only interested in finding one possible antecedent per anaphor, we took into account some search scope limitations.

8.1.1 Search scope

As a starting point for restricting the number of instances without losing possibly interesting information, we calculated the distance between the references and their immediately preceding antecedent. For these calculations, we took the MUC-6 and KNACK-2002 training sets as a test case. The distances were calculated as follows: antecedents from the same sentence as the anaphor were at distance 0. Antecedents in the sentence preceding the sentence of the referring expression, were at distance 1, and so on. We divided the group of referring expressions into the three categories: (1) pronouns, (2) proper nouns and (3) common nouns. These results are displayed in Figure 8.1 and Figure 8.2. Both figures reveal similar tendencies. With respect to the pronominal anaphors, we can observe that in the MUC-6 training data 97.8% of the antecedents appears in a context of three sentences. From these antecedents, the large majority (73.0%) appears in the sentence itself, 22.7% appears one sentence before and 2.2% of the antecedents of anaphorical pronouns is located two sentences before. In the KNACK-2002 training data for the pronouns, a similar but less prominent observation can be made. 77.3% of the immediately preceding antecedents can be found in a context of three sentences. 41.1% of the antecedents appears in the sentence itself, 29.2% appears one sentence before and 6.9% of the antecedents of anaphoric pronouns is located two sentences before. With respect to the named entities, we can observe that 79.2% of the named entities in the MUC-6 training data occurs in a scope of three sentences. For KNACK-2002, 44.01% of the immediately preceding antecedents of the proper noun NPs can be found in a scope of three sentences. Finally, for the common noun NPs we

can observe that in the MUC-6 training data 73.3% occurs in a scope of three sentences. For KNACK-2002, 65.2% of the immediately preceding antecedents can be found in a scope of three sentences.

Although similar tendencies can be observed in both data sets, these tendencies are much more prominent in the MUC-6 data. This difference might be due to a difference in text style (magazine articles for KNACK-2002 as opposed to newspaper articles in MUC-6), a difference in text length (KNACK-2002 has longer texts) and typological differences between both languages.

We will use this search scope information in the construction of the test instances, for example by restricting the number of test instances in the pronouns data set to anaphors with antecedents at distance 0, 1 and 2 (as for example also in Mitkov (1998) and Yang et al. (2003)). We will return to this issue later. For a more elaborate discussion on search scope for English, we refer to Mitkov (2002).

A second motivation for restricting the number of antecedents on the basis of their distance to the anaphor, only for the pronouns, are the decreasing classifier results for the antecedents further away, as shown in for MUC-6 in Figure 8.3. This tendency can be observed for both classifiers. For the other data sets (proper nouns and common nouns), no general conclusion can be drawn concerning the dependency of performance on the distance of the candidate antecedent to the anaphor.

8.1.2 Resulting data sets

For the construction of the test instances, all NPs starting from the second NP in a text are considered a possible anaphor, whereas all preceding NPs are considered possible antecedents of the anaphor under consideration. Since this type of instance selection leads to an enormous number of instances, we take into account the search scope observations discussed earlier in some simple heuristics:

- **Pronouns**: all NPs in a context of 2 sentences before the pronominal NP are included in the test sets for the pronouns.
- **Proper nouns**: all NPs which partially match the proper nouns NP are included. For the non matching NPs, the search scope is restricted to two sentences.
- **Common nouns**: same selection as for the proper nouns.

Figure 8.1: Distance in number of sentences between a given referring expression and its immediately preceding antecedent in the MUC-6 training set.

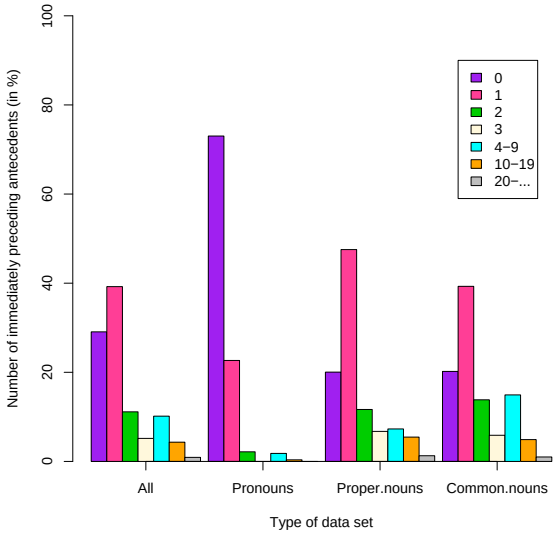


Figure 8.2: Distance in number of sentences between a given referring expression and its immediately preceding antecedent in the KNACK-2002 training set.

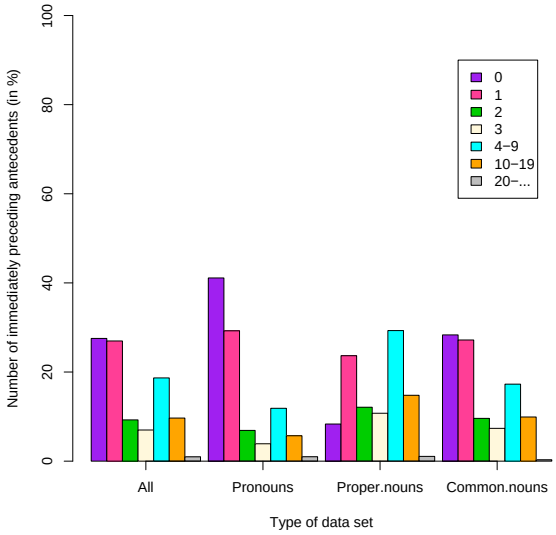
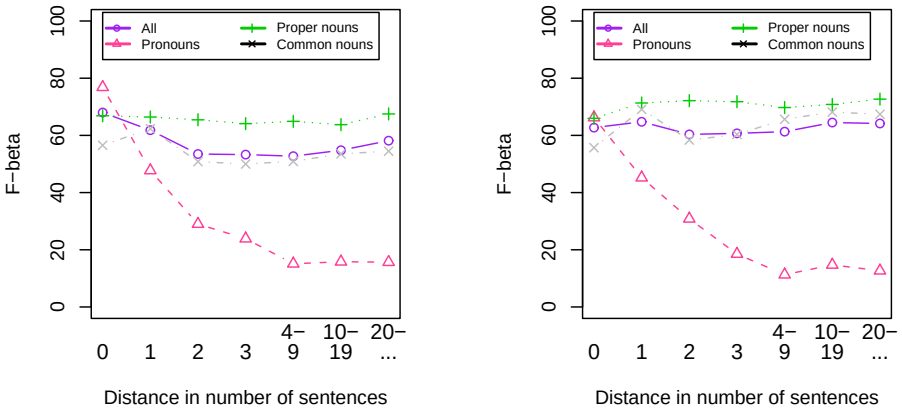


Figure 8.3: $F_{\beta=1}$ results plotted against the distance in number of sentences between an anaphor and its candidate antecedent after application of TIMBL (left) and RIPPER (right) on the MUC-6 data sets.



This instance selection allows us to obtain an overall test set reduction, including a large reduction of the number of negative instances.

8.2 Shift from the instance level to the coreference chain level

As already described in the previous section, our coreference resolution system involves a two-step procedure. In a first step, possibly coreferential NPs are classified as being coreferential or not. In a second step, the coreferential chains are built on the basis of the positively classified instances.

We can illustrate this testing procedure for the coreferential relation between "he" and "President Bush" in the following test sentence.

(39) **President Bush** met Verhofstadt in Brussels. **He** talked with our prime minister about the situation in the Middle East.

For the NP "he" test instances are built for the NP pairs displayed in Table 8.1.

After application of TIMBL or RIPPER, the result of the **first step** might be that the learner classifies the first instance as non-coreferential and the last two instances as being coreferential. Since we start from the assumption that each NP can only corefer with exactly one other preceding NP, a **second step** is required to make a choice between these two positive instances (he - Verhofstadt) and (he - President Bush).

Table 8.1: Test instances built for the "he" in example (39)

Antecedent	Anaphor	Classification
Brussels	he	no
Verhofstadt	he	yes
President Bush	he	yes

For this **second step**, different directions can be taken:

- Soon et al. (2001), for example, who use C4.5 as classifier, use a "closest-first" approach. They perform the selection as follows: for an anaphor j, the algorithm starts searching from the markable immediately preceding j and proceeds backward in the reverse order of the markables in the test document. The first markable found to be coreferent with the anaphor j (as determined by the trained decision tree classifier) is the antecedent.

- Ng and Cardie (2002a) perform single-link clustering to make clusters of coreferent NPs. They use a selection algorithm which performs a right-to-left search to find the most likely antecedent. This is done by selecting the antecedent with the highest confidence value among the candidate antecedents (all with class values above 0.5).

- Connolly, Burger and Day (1994) and Yang et al. (2003) use a so-called twin-candidate learning model, in which the antecedent for a given anaphor is identified as follows. For each set of possible antecedents for a given anaphor, they perform pairwise comparisons. Per comparison, the winner increases its score. The candidate antecedent with the maximal score is singled out as the antecedent for the given anaphor.

Instead of selecting one single antecedent per anaphor, as in the previously described approaches, we tried to build complete coreference chains for our documents. We will now continue with a description of our selection procedure.

8.2.1 Antecedent selection

We used the following counting mechanism to recover the coreference chains in the test documents.

1. Given an instance base with anaphor - antecedent pairs (ana_i, ant_{ij}), for which $i = 2$ to N and $j = i - 1$ to 0. Select all positive instances for each anaphoric NP. Then make groupings by adding the positive ant_{ij} to the group of ana_i and by adding ana_i to the group of ant_{ij}.

   The following is an example of such a grouping. The numbers represent IDs of anaphors/antecedents. The number before the colon is the ID of the anaphor/antecedent and the other numbers represent the IDs which relate to this anaphor/antecedent.

   2: 2 5 6 25 29 36 81 92 99 231 258 259 286
   5: 2 5 6 25 29 36 81 92 99 231 258 259 286
   6: 2 5 6 25 29 36 81 92 99 231 236 258 259 286
   8: 8 43 64 102 103 123 139 144 211 286
   20: 20 32 69 79

2. Then compare each ID grouping with the other ID groupings by looking for overlap between two groupings. Select the pairs with an overlap value above a predefined threshold. We selected all pairs with an overlap value above 0.1.

   E.g. If we consider the two first lines in the previous example, we can observe a complete overlap. Combining ID 8 with ID 2, however, leads to a very weak overlap (only on one ID) and an overlap value of 0.08. And no overlap is found for the combination of ID 20 and ID 2. If we take into account an overlap threshold of 0.1, this implies that the two last NP pairs in the table below will not be selected.

Overlap	ID+NP	ID+NP
1	5 Loral Space	2 Loral Space
0.08	8 Globalstar	2 Loral Space
0	20 Lockheed Martin Corp.	2 Loral Space

3. For each pair with an overlap value above the threshold, compute the union of these pairs. The following five lines show the coreferential chains for the proper nouns in the first document of the MUC-7 test set. These chains and their incremental construction are also represented in Table 8.2.1.

   8 43 44 64 102 103 123 139 144 211
   20 32 69 79

28 71 135 146 169 195 229 274
2 5 6 25 29 36 81 92 99 231 258 259 286
26 75 113 196

On top of this construction of coreferential chains, we also used some basic heuristics to select the most likely antecedent for a given anaphor. For the proper nouns data set, we preferred a complete match above a partial match. And for the common nouns we again preferred a complete match above a partial match and definite NPs above indefinite NPs.

8.2.2 Evaluation procedure

For all experiments reported on the training data, the performance was reported in terms of precision, recall and F-measure (as described in 4.4). For all experiments on the test set, the performance is also reported in terms of precision, recall and F-measure, but this time using the MUC scoring program from Vilain, Burger, Aberdeen, Connolly and Hirschman (1995). The program looks for the evaluation at equivalence classes, being the transitive closure of a coreference chain.

In the Vilain et al. (1995) algorithm, the **recall** for an entire set T of equivalence classes is computed as follows:

$$R_T = \frac{\sum(c(S)-m(S))}{\sum(c(S))}$$

where $c(S)$ is the minimal number of correct links necessary to generate the equivalence class S: $c(S) = (|S|-1)$. $m(S)$ is the number of missing links in the response relative to equivalence set S generated by the key: $m(S) = (|p(S)|-1)$. $p(S)$ is a partition of S relative to the response: each subset of S in the partition is formed by intersecting S and the responses sets R_i that overlap S. For the computation of the **precision**, the roles for the answer key and the response are reversed. For example, equivalence class S can consist of the following elements $S = \{1\ 2\ 3\ 4\}$. If the response is $< 1 - 2 >$, then $p(S)$ is $\{1\ 2\}$, $\{3\}$ and $\{4\}$.

This algorithm, however, has two major shortcomings according to Baldwin, Morton, Bagga, Baldridge, Chandraseker, Dimitriadis, Snyder and Wolska (1998). The algorithm does not give any credit for separating out singletons (entities occurring in chains only consisting of one element, such as 3 and 4 in the preceding example). Furthermore, it does not distinguish between different types

Table 8.2: Example output from the antecedent selection script for the proper nouns. The five tables show the incremental construction of the coreferential chains for the proper nouns in the first document of the MUC-7 test set (given in Appendix C). The last line of each table represents the resulting coreference chain.

ID + Anaphor <- ID + Antecedent
8 Globalstar <- 43 Globalstar Telecommunications Ltd.
8 Globalstar <- 43 Globalstar Telecommunications Ltd. <- 44 Globalstar <- 64 Globalstar
8 Globalstar <- 43 Globalstar Telecommunications Ltd. <- 44 Globalstar <- 64 Globalstar <- 102 Globalstar
8 Globalstar <- 43 Globalstar Telecommunications Ltd. <- 44 Globalstar <- 64 Globalstar <- 102 Globalstar <- 103 Globalstar
8 Globalstar <- 43 Globalstar Telecommunications Ltd. <- 44 Globalstar <- 64 Globalstar <- 102 Globalstar <- 103 Globalstar <- 123 Globalstar
8 Globalstar <- 43 Globalstar Telecommunications Ltd. <- 44 Globalstar <- 64 Globalstar <- 102 Globalstar <- 103 Globalstar <- 123 Globalstar <- 139 Globalstar
8 Globalstar <- 43 Globalstar Telecommunications Ltd. <- 44 Globalstar <- 64 Globalstar <- 102 Globalstar <- 103 Globalstar <- 123 Globalstar <- 139 Globalstar <- 144 Globalstar
8 Globalstar <- 43 Globalstar Telecommunications Ltd. <- 44 Globalstar <- 64 Globalstar <- 102 Globalstar <- 103 Globalstar <- 123 Globalstar <- 139 Globalstar <- 144 Globalstar <- 211 Globalstar

26 Monday <- 75 Monday
26 Monday <- 75 Monday <- 113 Monday
26 Monday <- 75 Monday <- 113 Monday <- 196 Monday

20 Lockheed Martin Corp. <- 32 Lockheed
20 Lockheed Martin Corp. <- 32 Lockheed <- 69 Lockheed
20 Lockheed Martin Corp. <- 32 Lockheed <- 69 Lockheed <- 79 Lockheed

2 Loral Space <- 5 Loral Space
2 Loral Space <- 5 Loral Space <- 6 Loral
2 Loral Space <- 5 Loral Space <- 6 Loral <- 25 Loral Corp.
2 Loral Space <- 5 Loral Space <- 6 Loral <- 25 Loral Corp. <- 29 Loral
2 Loral Space <- 5 Loral Space <- 6 Loral <- 25 Loral Corp. <- 29 Loral <- 36 Loral
2 Loral Space <- 5 Loral Space <- 6 Loral <- 25 Loral Corp. <- 29 Loral <- 36 Loral <- 81 Loral
2 Loral Space <- 5 Loral Space <- 6 Loral <- 25 Loral Corp. <- 29 Loral <- 36 Loral <- 81 Loral <- 92 Loral Space and Communications Corp.
2 Loral Space <- 5 Loral Space <- 6 Loral <- 25 Loral Corp. <- 29 Loral <- 36 Loral <- 81 Loral <- 92 Loral Space and Communications Corp. <- 99 Loral
2 Loral Space <- 5 Loral Space <- 6 Loral <- 25 Loral Corp. <- 29 Loral <- 36 Loral <- 81 Loral <- 92 Loral Space and Communications Corp. <- 99 Loral <- 231 Loral Space
2 Loral Space <- 5 Loral Space <- 6 Loral <- 25 Loral Corp. <- 29 Loral <- 36 Loral <- 81 Loral <- 92 Loral Space and Communications Corp. <- 99 Loral <- 258 Loral Space
2 Loral Space <- 5 Loral Space <- 6 Loral <- 25 Loral Corp. <- 29 Loral <- 36 Loral <- 81 Loral <- 92 Loral Space and Communications Corp. <- 99 Loral <- 231 Loral Space <- 258 Loral Space <- 259 Loral
2 Loral Space <- 5 Loral Space <- 6 Loral <- 25 Loral Corp. <- 29 Loral <- 36 Loral <- 81 Loral <- 92 Loral Space and Communications Corp. <- 99 Loral <- 231 Loral Space <- 258 Loral Space <- 259 Loral <- 286 Loral Space and Globalstar

28 Bernard Schwartz <- 71 Schwartz
28 Bernard Schwartz <- 71 Schwartz <- 135 Bernard Schwartz
28 Bernard Schwartz <- 71 Schwartz <- 135 Bernard Schwartz <- 146 Bernard Schwartz
28 Bernard Schwartz <- 71 Schwartz <- 135 Bernard Schwartz <- 146 Bernard Schwartz <- 169 Schwartz
28 Bernard Schwartz <- 71 Schwartz <- 135 Bernard Schwartz <- 146 Bernard Schwartz <- 169 Schwartz <- 195 Schwartz
28 Bernard Schwartz <- 71 Schwartz <- 135 Bernard Schwartz <- 146 Bernard Schwartz <- 169 Schwartz <- 195 Schwartz <- 229 Schwartz
28 Bernard Schwartz <- 71 Schwartz <- 135 Bernard Schwartz <- 146 Bernard Schwartz <- 169 Schwartz <- 195 Schwartz <- 229 Schwartz <- 274 Schwartz

of errors. In the following example, the key consists of three equivalence classes and two responses are given. The two responses yield the same precision (88.9%) and recall score (100%) according to the MUC scoring program. Baldwin et al. (1998) argue that the error made in response 2 is more damaging, since it makes more entities erroneously coreferent.

key	0←1←2←3
	4←5
	6←7←8←9←10
response 1	0←1←2←3←4←5
	6←7←8←9←10
response 2	0←1←2←3←6←7←8←9←10
	4←5

Despite these shortcomings, we used this scoring software since it has been widely used for evaluation on the MUC-6 and MUC-7 data sets and it thus enables comparison with the results of other systems on these data sets.

8.3 Experimental results

In this section, we will report the results on the MUC-6, MUC-7 and KNACK-2002 test data sets. In order to evaluate the performance of our classifiers, we also calculated two baseline scores.

- **Baseline I**: For the calculation of the first baseline, we did not take into account any linguistic, semantic or location information. This implies that this baseline is calculated on the large test corpus which links every NP to every preceding NP and not on the smaller test corpora described in Section 8.1.2 which already take into account feature information. Baseline I is obtained by linking every noun phrase to its immediately preceding noun phrase.
- **Baseline II**: The second, somewhat more sophisticated baseline is the result of the application of the following simple, common-sense rules: select the closest antecedent with the same gender and number (for the pronouns data set), select the closest antecedent which partially/completely matches the NP (for the proper and common nouns data sets).

Table 8.3 shows the precision, recall and $F_{\beta=1}$ scores for these two baselines. It reveals the following tendencies. Linking every NP to the immediately preceding NP, as was done for the first baseline, leads to high overall recall scores:

81.5% for MUC-6, 73.8% for MUC-7 and 81.9% for KNACK-2002. The precision scores, on the other hand, are low: 30.7% for MUC-6, 29.3% for MUC-7 and 27.9% for KNACK-2002. The Baseline II scores which depend on feature information, are more balanced: 58.8% recall and 43.4% precision for MUC-6, 48.4% recall and 43.5% precision for MUC-7 and 45.7% recall and 38.9% precision for KNACK-2002. The highest $F_{\beta=1}$ values are obtained by Baseline II: 49.9% on the MUC-6 test set, 45.8% on the MUC-7 test data and 42.0% for KNACK-2002.

With respect to the baseline results on the NP type data sets, the following observations can be made. The Baseline I results on the "Pronouns", "Proper nouns" and "Common nouns" data sets are all very low. The largest part of the scores is below 10%. An exception are the precision scores for the pronouns: 25.2% for MUC-6, 22.2% for MUC-7 and 18.1% for KNACK-2002. These results once again confirm that the antecedent of a pronominal anaphor is located close to the anaphor, as already shown in Section 8.1. With respect to the Baseline II results, we can conclude that the application of some simple rules leads to an increase of precision compared to the Baseline I results.

8.3.1 Classifier results

In the previous chapter, we optimized our classifiers on the "Pronouns", "Proper nouns" and "Common nouns" training sets. Therefore, we decided to train the optimized TIMBL and RIPPER on the NP type training data and to test them also on the corresponding test sets. Table 8.4 gives an overview of the results obtained by TIMBL and RIPPER on the "Pronouns", "Proper nouns" and "Common nouns" test sets in terms of precision, recall and $F_{\beta=1}$.

Table 8.4 shows that our results obtained on the MUC-6 and MUC-7 data sets are comparable to the results reported by Soon et al. (2001). They report a precision of 67.3%, a recall of 58.6% and an $F_{\beta=1}$ of 62.6% on the MUC-6 data. For MUC-7, they report a precision of 65.5%, a recall of 56.1% and an $F_{\beta=1}$ of 60.4%. The best results reported to date on the MUC-6 and MUC-7 data are by Ng and Cardie (2002a,2002b,2002c): 63.3% recall, 76.9% precision and 69.5% F_β on MUC-6 and 54.2% recall, 76.3% precision and 63.4% F_β on MUC-7[1] Their extensions to the approach of Soon et al. (2001) include (i) the expansion of the feature set with additional lexical, semantic and knowledge-based features, (ii) a modification of the clustering algorithm favoring the 'highest likely antecedent', (ii) a learning-based method to determine anaphoricity, (iii) positive and negative sample selection in order to handle the problem of skewed

[1] On MUC-6, they report a top performance of 70.4% F_β when doing manual feature selection.

Table 8.3: A baseline score for the different test data sets. The recall and $F_{\beta=1}$ scores could not be provided for the NP type data sets, since the scoring software does not distinguish between the three NP types.

MUC-6	Prec.	Rec.	$F_{\beta=1}$
Baseline I			
PPC	30.7	**81.5**	44.6
Pronouns	25.2	—	—
Proper nouns	4.6	—	—
Common nouns	7.4	—	—
Baseline II			
PPC	**43.4**	58.8	**49.9**
Pronouns	55.8	—	—
Proper nouns	53.0	—	—
Common nouns	30.3	—	—

MUC-7	Prec.	Rec.	$F_{\beta=1}$
Baseline I			
PPC	29.3	**73.8**	42.0
Pronouns	22.2	—	—
Proper nouns	5.1	—	—
Common nouns	6.2	—	—
Baseline II			
PPC	**43.5**	48.8	**45.8**
Pronouns	47.5	—	—
Proper nouns	48.8	—	—
Common nouns	33.1	—	—

KNACK-2002	Prec.	Rec.	$F_{\beta=1}$
Baseline I			
PPC	27.9	**81.9**	41.7
Pronouns	18.1	—	—
Proper nouns	2.4	—	—
Common nouns	4.9	—	—
Baseline II			
PPC	**38.9**	45.7	**42.0**
Pronouns	39.2	—	—
Proper nouns	56.9	—	—
Common nouns	23.6	—	—

class distributions, (iv) pruning of the rule sets, etc.

For KNACK-2002, no comparative results are yet available since this is a new corpus. Table 8.4 shows that both TIMBL and RIPPER obtain a $F_{\beta=1}$ score of 51% on the Dutch data. As for English, the precision scores for the "Pronouns" (64.9% for TIMBL and 66.7% for RIPPER) and the "Proper nouns" data sets (79.4% for TIMBL and 79.0% for RIPPER) are much higher than those obtained on the "Common nouns" data set (47.6% for TIMBL and 47.5% for RIPPER). Furthermore, the recall scores are about 20% lower than the precision scores: 42.2% recall vs. 65.9% precision for TIMBL and 40.9% recall vs. 66.3% precision for RIPPER.

8.4 Error analysis

In order to discover regularities in the errors committed by TIMBL and RIPPER, we performed a manual error analysis on three MUC-7 and three KNACK-2002 documents. For both data sets, we selected one document on which our system performs above average and two documents for which the $F_{\beta=1}$ score is below average. In each of these documents, we looked for the errors committed by the "Pronouns", "Proper nouns" and "Common nouns" learning modules. We divided these errors into two groups, according to the scoring scheme: recall errors and precision errors. The recall errors are caused by classifying positive instances as being negative. These false negatives cause missing links in the coreferential chains. The precision errors, on the other hand, are caused by classifying negative instances as being positive. These false positives cause spurious links in the coreferential chains. Table 8.5 and Table 8.6 give an overview of these precision, recall and $F_{\beta=1}$ errors per document and per NP type data set. We will now discuss some of the precision and recall errors in more detail for all three types of NPs in both data sets. We start off with the errors made on the MUC-7 data.

8.4.1 MUC-7

Table 8.5 gives an overview of the precision, recall and $F_{\beta=1}$ errors made by TIMBL on three MUC-7 test documents (given in Appendix C). It lists the errors per document and per NP type data set. It shows that the "Proper nouns" learning module always gives high precision and recall scores. For the "Pronouns" data sets, high recall and precision scores can be observed for the first two test documents. These scores drop considerably for the third data set. The results for the "Common nouns" data sets are considerably lower than

Table 8.4: Results from TIMBL and RIPPER on the test set in terms of precision, recall and $F_{\beta=1}$. No recall and $F_{\beta=1}$ scores could be provided on the NP type data sets, since the scoring software does not distinguish between the three NP types.

MUC-6	Prec.	Rec.	$F_{\beta=1}$
Timbl			
PPC	70.5	59.1	64.3
Pronouns	77.3	—	—
Proper nouns	83.0	—	—
Common nouns	56.4	—	—
Ripper			
PPC	66.2	60.9	63.4
Pronouns	79.9	—	—
Proper nouns	82.2	—	—
Common nouns	50.4	—	—

MUC-7	Prec.	Rec.	$F_{\beta=1}$
Timbl			
PPC	54.5	67.1	60.2
Pronouns	68.4	—	—
Proper nouns	78.0	—	—
Common nouns	54.3	—	—
Ripper			
PPC	68.7	49.5	57.6
Pronouns	67.8	—	—
Proper nouns	82.5	—	—
Common nouns	57.2	—	—

KNACK-2002	Prec.	Rec.	$F_{\beta=1}$
Timbl			
PPC	65.9	42.2	51.4
Pronouns	64.9	—	—
Proper nouns	79.4	—	—
Common nouns	47.6	—	—
Ripper			
PPC	66.3	40.9	50.6
Pronouns	66.7	—	—
Proper nouns	79.0	—	—
Common nouns	47.5	—	—

those for the other two data sets, both on the precision and the recall side. We will now discuss some of the precision and recall errors in more detail for the three types of NPs.

Pronominal missing and spurious links

In all of the following sentences, the "Pronoun" resolution system for the MUC-7 test data, has either missed a coreferential link (40, 41) or created a spurious link (45, 47).

(40) In addition, Schwartz said **Loral Space** would use **its** holdings in Space Systems Loral, a private maker of satellites, to expand into the direct broadcast satellite business. "Any service that is based on satellites is going to be a fertile area for **our** investment," he said. (document 9601080668)

The "our" in sentence (40) represents a missing link. Due to the lack of matching, gender agreement, number agreement, etc. the instances made for this pronoun do not contain enough evidence for linking this pronoun to any of the preceding noun phrases.

(41) The company's 11 1/2-year-old Silicon Studio subsidiary will work with Sega Enterprises of Japan, SegaSoft and Time Warner Interactive, among others, to test **the software**. **It** will be sold starting this summer. (document 9602290649)

The "It" is classified as being not coreferential with any of the preceding NPs.

Nearly half of the missing links (23 out of 51 missing links for MUC-7) involve the pronoun "it". The current features do not allow us to distinguish between anaphoric and pleonastic pronouns. In the following three sentences from the MUC-7 test document 9601160264, for example, the pronoun "it" occurs two times in a pleonastic way (43, 44) and once as an anaphor (42).

(42) "**Satellites** give us an opportunity to increase the number of customers we are able to satisfy with the McDonald's brand,"said McDonald's Chief Financial Officer, Jack Greenberg. "**It**'s a tool in our overall convenience strategy."

Table 8.5: Number of precision, recall and $F_{\beta=1}$ errors for three MUC-7 test documents. The errors are calculated per NP type data set. The three test documents are provided in Appendix C.

MUC-7 document 9601080668	Precision	Recall	$F_{\beta=1}$
Pronouns	8/8	8/11	
	100.00%	72.73%	84.21%
Proper nouns	32/39	32/34	
	82.05%	94.12%	87.67%
Common nouns	9/19	9/17	
	47.37%	52.94%	50.00%

MUC-7 document 9602290649	Precision	Recall	$F_{\beta=1}$
Pronouns	3/4	3/5	
	75.00%	60.00%	66.67%
Proper nouns	13/17	13/13	
	76.47%	100.00%	86.67%
Common nouns	10/31	10/18	
	32.26%	55.56%	40.82%

MUC-7 document 9609100495	Precision	Recall	$F_{\beta=1}$
Pronouns	1/3	1/3	
	33.33%	33.33%	33.3%
Proper nouns	10/11	10/14	
	90.91%	71.43%	80.00%
Common nouns	7/21	7/12	
	33.33%	58.33%	42.42%

(43) "**It**'s been good for both companies," said Buddy Burns, Wal-Mart's manager of branded food service.

(44) "**It** adds to the overall shopping experience to have McDonald's there."

The "Pronoun" resolution system does not only miss certain links. It also created some spurious links, yet in a much lesser degree. In sentence (45), for example, our resolution system erroneously (according to the MUC annotation) links the second "you" to the first "you" since the MUC annotation considers both pronouns as being pleonastic. In sentence (46), however, all three "you(r)" pronouns are annotated as being coreferential and they are also correctly resolved.

(45) They want to be the first sign **you** see when **you** get hungry," said Dennis Lombardi, an analyst at Chicago-based market researcher Technomics Inc. (document 9601160264)

(46) "**You** have to pick **your** partners pretty carefully because they may not keep up to **your** standards," said Technomics' Lombardi. (document 9601160264)

In short, a large part of the errors made by the pronominal resolution system involves the false distinction between anaphoric and pleonastic pronouns. Although it is not always clear to us how this distinction has been made in the MUC-annotation scheme (e.g. in example 45 and 46), more effort should be put in features which can capture this difference between anaphoric and pleonastic pronouns. Mitkov, Evans and Orasan (2002), for example, use TIMBL to automatically classify instances of "it" as pleonastic or nominal anaphora. They report an overall classification rate of 78.74% using ten-fold cross-validation.

Furthermore, we also observed that "it" often links to a false preceding NP, such as in the following example.

(47) **Hughes Electronics Corp.** has paid the U.S.government $ 4 million to settle a 1990 lawsuit filed by two former employees who accused **it** of lying to the Pentagon, the Justice Department said. (document 9609100495)

After TIMBL classification, the "it" is linked to three preceding NPs: "U.S", "the U.S. government" and "Hughes Electronics Corp." Picking the nearest NP, namely "the U.S. government" as antecedent leads to an erroneous link. It is however not clear to us yet how this type of classification errors can be avoided without a representation of world knowledge.

Missing and spurious links made for the proper nouns

Although Table 8.5 shows high recall and precision scores for the proper nouns, there is still room for improvement. Errors on the proper nouns are caused by errors in NP chunking (48), errors in the recognition of phrase embedding (50), errors in part of speech tagging (49), etc. We will now discuss some of these errors.

(48) **Hughes Electronics Corp.** has paid the U.S. government $ 4 million to settle a 1990 lawsuit filed by two former employees who accused it of lying to the Pentagon, the Justice Department said. (...) The fine, which settles a civil case related to the charges, is the second **Hughes** has paid because of that evidence. (document 9609100495)

Our system does not make a link between "Hughes" and "Hughes Electronics Corp.", due to an NP chunking error. "the second Hughes" is considered as one single NP, instead of two NPs.

(49) If **Globalstar** begins its service on schedule in 1998, he predicted that the company would have 3 million customers by 2,002. **Globalstar** still needs to raise $ 600 million, and Schwartz said that the company would try to raise the money in the debt market. (document 9601080668)

In this example, our system misses the link between the two "Globalstar" NPs. This missing links is caused by a false part-of-speech tag for the second "Globalstar". Since this second "Globalstar" is not tagged as an NP, it is not selected for coreference resolution.

(50) **Hughes** pays U.S. $ 4 mln fine from whistleblower case. (...) **Hughes Electronics Corp.** has paid the U.S. government $ 4 million to settle a 1990 lawsuit filed by two former employees who accused it of lying to the Pentagon, the Justice Department said. (document 9609100495)

On top of the correct link between "Hughes Electronics Corp." and "Hughes", our system also creates a spurious link for "Hughes" in the larger NP "Hughes Electronics Corp.".

If we consider the output of our resolution system for the proper nouns in document 9601080668, as shown in the tables of Section 8.2, it reveals the strong importance of string matching for coreference resolution of proper nouns. Other crucial features for the resolution of this type of coreferences are the apposition

feature (as in the link "McDonald's Chief Financial Officer <- Jack Greenberg") and the alias feature (as in the link "Silicon Graphics Inc. <- SGI").

Missing and spurious links made for the common noun NPs

Although there is still room for improvement both for the resolution of pronominal coreferences and the resolution of coreferential relations between proper noun NPs, the resolution of the anaphoric common noun NPs presents the greatest challenge, as shown in Tables 8.4 and 8.5.

Most of the errors discussed below reveal that with the current features this type of coreferences cannot be captured. A better performing common noun resolution would require both a deeper semantic and syntactic analysis. As for the coreference resolution of anaphoric proper nouns, missing links can be caused by preprocessing errors, such as part-of-speech tagging, NP chunking, apposition recognition, etc. Other errors are typical for the resolution of coreferential relations between common nouns. We will now discuss some of the errors which are typically made by the common noun resolution module.

(51) One reason Lockheed Martin Corp. did not announce a full acquisition of Loral Corp. on Monday, according to Bernard Schwartz, Loral's chairman, was that Lockheed could not meet the price he had placed on **Loral's 31 percent ownership of Globalstar Telecommunications Ltd.** (...) News of Monday's deal, in which Lockheed will buy most of Loral's military businesses and invest $ 344 million in Loral Space and Communications Corp., a new company whose **principal holding** will be **Loral's interest in Globalstar**, sent Globalstar's own shares soaring $6.375, to $40.50 in Nasdaq trading. (document 9601080668)

Resolving the links in this example requires both a deeper semantic and syntactic analysis. Through the use of more fine-grained semantic features, "ownership" and "interest" should be linked to each other. This could be done by the more fine-grained use of a semantic lexicon such as WordNet or by first performing word-sense disambiguation on these words. Furthermore, an extra feature should be included which denotes the type of verb (e.g. linking verb) between two NPs. A similar feature should also capture the relations in sentences (52) and (53).

(52) Though the idea of setting up a global telephone network based on dozens of satellites appears the stuff of science fiction, Schwartz and many others, including Motorola Inc., several international telecommunications companies and William Gates, the chairman of Microsoft

Corp., see **it** as **a very real opportunity**. (document 9601080668)

The MUC annotation links "a very real opportunity" to "it". But the annotation does not contain the, in our perception, similar link "the stuff of science fiction" to "the idea of setting up a global telephone network based on dozens of satellites".

(53) If Globalstar begins its service on schedule in 1998, he predicted that the company would have 3 million customers by 2,002 , bringing in **$ 2.7 billion** in **annual revenue**. (document 9601080668)

Some additional examples of missing links are shown in sentences (54), (55) and (56).

(54) On Monday, industry sources said, Mountain View-based Silicon Graphics Inc. will release a technology dubbed **" FireWalker "** designed to make the next generation of video games with 3-D images more economical and commonplace. The company's 11/2-year-old Silicon Studio subsidiary will work with Sega Enterprises of Japan, SegaSoft and Time Warner Interactive, among others, to test the **software**. (...) San Francisco's Rocket Science plans to release the first video game using the **technology** by Christmas. (document 9602290649)

(55) Hughes pays U.S. $ 4 mln fine from **whistleblower case**. Hughes Electronics Corp. has paid the U.S. government $ 4 million to settle a **1990 lawsuit**.(document 9609100495)

(56) China's Foreign Trade Minister Wu Yi has extended an olive branch to **Taiwan** saying Beijing remained committed in talks with the **" breakaway island "** to establish direct trade and communication links.

8.4.2 KNACK-2002

Table 8.6 gives an overview of the precision, recall and $F_{\beta=1}$ errors made by TIMBL on three KNACK-2002 test documents (provided in Appendix D). It lists the errors per document and per NP type data set. It shows that the "Proper nouns" learning module yields the highest precision and recall scores. The results for the "Common nouns" data sets are considerably lower than those for the two other data sets, both on the precision and the recall side. We will now discuss some of the precision and recall errors in more detail for the three types of NPs in three KNACK-2002 documents.

Table 8.6: Number of precision, recall and $F_{\beta=1}$ errors for three KNACK-2002 test documents. The errors are calculated per NP type data set. The three KNACK-2002 test documents are provided in Appendix D.

KNACK-2002 document 1	Precision	Recall	$F_{\beta=1}$
Pronouns	2/4	2/3	
	50.00%	66.67%	57.14%
Proper nouns	4/4	4/6	
	100.00%	66.67%	80.00%
Common nouns	0/0	0/2	
	0.00%	0.00%	0.00%

KNACK-2002 document 2	Precision	Recall	$F_{\beta=1}$
Pronouns	1/2	1/5	
	50.00%	20.00%	28.57%
Proper nouns	3/3	3/4	
	100.00%	75.00%	85.71%
Common nouns	2/2	2/6	
	100.00%	33.33%	50.00%

KNACK-2002 document 3	Precision	Recall	$F_{\beta=1}$
Pronouns	3/3	3/6	
	100.00%	50.00%	66.67%
Proper nouns	0/0	0/1	
	0.00%	0.00%	0.00%
Common nouns	1/1	1/7	
	100.00%	14.29%	25.01%

Pronominal missing and spurious links

In all of the following sentences, the "Pronoun" resolution system for the KNACK-2002 test data, has either missed a coreferential link (57, 58 and 59) or created a spurious link (60).

(57) Piqué wilde **de onbemande camera's** niet homologeren omdat **ze** 'niet voldeden aan de opgelegde normen'.

English: Piqué refused to homologate **the unmanned cameras** since '**they** did not comply with the desired standards'. (document 2)

(58) **De moeder van Moussaoui** gaf een persconferentie waarin **ze** om een eerlijk proces vroeg.

English: **The mother of Moussaoui** gave a press conference in which **she** asked for a fair trial. (document 3)

(59) Stevaert ergert zich aan de manier waarop **de verschillende ministeries** het dossier naar **elkaar** toeschuiven.

English: Stevaert gets annoyed about the way **the different ministries** pass **each other** the file. (document 2)

(60) In de opiniepeilingen liggen Jospin en Chirac **zij** aan **zij**.

English: In the polls, Jospin and Chirac are **side** by **side**. (document 1)

The missing links for the pronouns in (57) and (59) are caused by the lack of evidence for a positive classification in the feature vectors. The missing link in (58) is caused by a false part-of-speech tag: the "ze" is tagged as a third person plural pronoun. The same part-of-speech tag is also given to the two occurrences of the common noun "zij" in (60) causing a spurious link.

Furthermore, a large part of the errors made for "het" and "dat" involve the false distinction between anaphoric and pleonastic pronouns, as shown in (61) as opposed to a correct (62).

(61) Een god van **het vuur**. Als vice-minister van Defensie heeft Paul Wolfowitz alles bij elkaar eigenlijk een bescheiden job in de Amerikaanse regering. Hoe komt **het** dan dat hij zoveel invloed heeft in het Witte Huis?

English: A god of **the fire**. As a vice minister of Defense, Paul Wolfowitz in the end has a rather insignificant job in the American government. How is **it** possible that he has so much influence in the White House?

(62) Maar voorzitter Spiritus-Dassesse gelooft niet in **het nieuwe plan**. **Het** lijkt te veel op het vorige.

English: But chairwoman Spiritus-Dassesse does not have faith in **the new plan**. **It** resembles the previous one too much.

In short, the resolution of the pronominal anaphors is hindered by the low informativeness of some feature vectors which do not allow for distinction between positive and negative classification, by the lack of features which can capture the pleonastic and anaphoric use of pronouns and by preprocessing errors, mainly part-of-speech tagging errors. Furthermore, for the Dutch male and female pronouns, such as "hij", "hem", "haar", the search space of candidate antecedents is much larger than that for the corresponding English pronouns, since they can also refer to the linguistic gender of their antecedent. An example of this is given in (63).

(63) Zij stelden boudweg dat het moeilijk zou zijn om **de studie** te 'dupliceren'. Waarmee eigenlijk werd gezegd dat **ze** niet wetenschappelijk verantwoord was uitgevoerd.

English: They boldly argued that it would be hard to 'duplicate' **the study**. By which was claimed that **it (Dutch: "she")** was not carried out in a scientific way.

Missing and spurious links made for the proper nouns

Although high recall and precision scores can be observed for the proper nouns, there is still room for improvement. As for English, the errors on the proper nouns are mainly caused by preprocessing errors: errors in NP chunking and errors in part of speech tagging (64, 65), etc. We will now discuss some of these errors. The part-of-speech tagger trained on the Spoken Dutch Corpus mainly assigns three different types of tags to proper nouns: SPEC(deeleigen) (as for "Zacarias Moussaoui" and "Charles Picqué"), SPEC(afgebr) (as for "Moussaoui") and "N(eigen (...)" (as for "Stevaert"). The corresponding chunks for the underlying part-of-speech tags are "MWU" (multi word unit) for SPEC(deeleigen) and SPEC(afgebr) and "NP" for "N(eigen (...)". Since multi word units can also consist of non-NP combinations (e.g. "in staat"), these multi word units are not always selected for resolution.

(64) **Zacarias Moussaoui**, de eerste persoon die door het Amerikaanse gerecht aangeklaagd is voor de terreuraanvallen van 11 september, pleit onschuldig bij zijn eerste verschijning voor de rechtbank. (...) De moeder van **Moussaoui** vloog enige dagen voor zijn voorleiding naar de Verenigde Staten.

English: **Zacarias Moussaoui**, the first person who has been charged by the American judicial authorities for the terrorist attacks of 11 September, pleads not guilty at the first hearing. (...) The mother of **Moussaoui** came to the United States a few days before the hearing. (document 3)

(65) Donderdag gaven Stevaert en **Charles Picqué** (PS), federaal minister van Economische Zaken, elkaar de schuld voor het disfunctioneren van twee onbemande camera's op de A12 in Willebroek. **Picqué** - bevoegd voor de erkenning van de flitspalen - wilde de onbemande camera's niet homologeren omdat ze 'niet voldeden aan de opgelegde normen'.

English: On Thursday, Stevaert and **Charles Picqué** (PS), federal secretary of Economic Affairs, blamed each other for the disfunctioning of two unmanned cameras at the A12 in Willebroek. **Picqué** - authorized for the homologation of the flash-guns - refused to homologate the unmanned cameras since 'they did not comply with the desired standards'. (document 2)

Missing and spurious links made for the common noun NPs

Both Tables 8.4 and 8.6 show that the resolution of coreferential relations between common noun NPs is problematic. Most of the errors discussed below reveal that with the current features this type of coreferences cannot be captured. As also stated for the English data, a better performing common noun resolution would require both a deeper semantic and syntactic analysis. As for the coreference resolution of anaphoric proper nouns and pronouns, missing links can be caused by preprocessing errors, such as part-of-speech tagging, NP chunking, apposition recognition, the recognition of subjects, objects and predicates (66), etc. Other errors are typical for the resolution of coreferential relations between common nouns: the lack of recognizing synonyms as in (68), the lack of recognizing hyponyms as in (69), or in other words ... the lack of world knowledge. We will now illustrate some of the errors which are typically made by the common noun resolution module.

(66) **De socialist Jospin en de gaullist Chirac** zijn **de belangrijkste kandiaten voor het hoogste ambt**.

English: **The socialist Jospin and the gaullist Chirac** are **the most important candidates for the highest office**. (document 1)

(67) Vlaams minister van Mobiliteit Steve Stevaert dreigt met een regeringscrisis als de federale regering blijft weigeren mee te werken aan **het verbeteren van de verkeersveiligheid**. (...) Stevaert ergert zich aan de manier waarop de verschillende ministeries **het dossier** naar elkaar toeschuiven.

English: Flemish Minister for Mobility Steve Stevaert threatens with a government crisis if the federal government keeps on refusing to cooperate for **an improvement of traffic safety**. (...) Stevaert gets annoyed about the way the different ministries shift **the file** on to each other. (document 2)

(68) Donderdag gaven Stevaert en Charles Picqué elkaar de schuld voor het disfunctioneren van **twee onbemande camera's** op de A12 in Willebroek. Picqué - bevoegd voor de erkenning van **de flitspalen** - (...)

English: On Thursday, Stevaert and Charles Picqué (PS) blamed each other for the disfunctioning of **two unmanned cameras** at the A12 in Willebroek. **Picqué** - authorized for the homologation of **the flashguns** - (...) (document 2)

(69) **Zacarias Moussaoui**, de eerste persoon die door het Amerikaanse gerecht aangeklaagd is voor **de terreuraanvallen van 11 september**, pleit onschuldig bij zijn eerste verschijning voor de rechtbank. (...) De Fransman van Marokkaanse afkomst wordt ervan verdacht de 'twintigste vliegtuigkaper' te zijn die door omstandigheden niet aan **de kapingen** kon deelnemen.

English: Zacarias Moussaoui, the first person who has been charged by the American judicial authorities for **the terrorist attacks of 11 September**, pleads not guilty at the first hearing. (...) **The French citizen of Moroccan descent** is accused of being the 'twentieth hijacker' who was prevented from carrying out **the hijackings**. (document 3)

A spurious link is created in the following example:

(70) Wolfowitz speelde een grote rol in de manier waarop de Verenigde Staten **de Golfoorlog** aanpakten . (...) Als dat land nu boven aan de lijst staat van landen die het volgende doelwit in **de oorlog** tegen het terrorisme zouden kunnen worden, (...)

English: Wolfowitz had a key role in the way the United States conducted **the Gulf war**. (...) If that country is at the top of the list of the countries which may be the next target in **the war** against terror (...)

8.5 Summary

In this chapter, we reported the results on the MUC-6, MUC-7 and KNACK-2002 test sets. Contrary to the previous chapters, in which results on the instance level were reported, the results in this chapter are reported for the coreference chains. On all data sets, we showed that there is substantial room for improvement. Through a qualitative error analysis on three English and on three Dutch texts, we illustrated what caused missing and spurious links.

On the basis of the error analysis, the following observations could be made. Independently of the type of anaphor, we could observe that preprocessing errors, such as errors in part-of-speech tagging, NP chunking and relation finding, caused a large number of errors. For Dutch, we could observe that this error load of preprocessing was highly damaging (e.g. for the pronominal "ze" and for the proper nouns). With respect to the resolution of pronominal anaphors, we observed for both Dutch and English that the current features do not allow us to distinguish between anaphoric and pleonastic pronouns. For Dutch, we could also observe that for the male and female pronouns, the search space of possible candidate antecedents was much larger than for English, since they can also refer to the linguistic gender of the antecedent. With respect to the proper noun anaphors, we could observe that the errors were mainly caused by preprocessing errors, and also errors in alias detection. Finally, with respect to the errors typically made for the resolution of common noun anaphors we saw that the current shallow features could not capture a large number of coreferential links. Furthermore, we could observe for Dutch that the construction of the semantic features was hindered by a too restricted Dutch EuroWordNet. A global observation for both languages is that a deeper semantic and syntactic analysis is required. We will return to these errors in the following concluding chapter discussing future work.

CHAPTER 9

Conclusion

In this thesis, we presented a machine learning approach to the resolution of coreferential relations between nominal constituents in Dutch. It is the first automatic resolution approach proposed for this language. In order to enable a corpus-based strategy, we first annotated a corpus of Dutch news magazine text, KNACK-2002, with coreferential information for pronominal, proper noun and common noun coreferences. A separate learning module was built for each of these NP types. The main motivation for this approach was that the information sources which are important for the resolution of the coreferential relations differ per NP type. This approach was not only applied to Dutch, for which no comparative results are yet available, but also to the well-known English MUC-6 and MUC-7 data sets.

Coreference and the task of coreference resolution was the main point of interest in Chapters 2 and 3 and in Chapter 8 on testing. In the chapters in between, we focused on the methodological issues which arise when performing a machine learning of coreference resolution experiment, or more broadly, a machine learning of language experiment. In the following two sections, we discuss the main observations from the research questions formulated in Section 1.3.

9.1 Methodological issues: main observations

Algorithm 'bias' In Chapter 4 we investigated the effect of algorithm 'bias' on learning coreference resolution. This was done because to our knowledge, this effect of 'bias' was not yet systematically investigated in the existing machine learning of coreference resolution literature. The existing machine learning approaches to coreference resolution use the C4.5 decision tree learner (Quinlan 1993), used by Aone and Bennett (1995), McCarthy (1996) and Soon et al. (2001), maximum entropy learning as in Yang et al. (2003) or the RIPPER rule learner (Cohen 1995) as in Ng and Cardie (2002a;2002b;2002c). By contrasting the performance of two completely different learning techniques, namely memory-based learning and rule induction, on this task of coreference resolution, we wanted to determine the effect of algorithm 'bias' on learning coreference resolution. Two machine learning packages were used in the experiments: the memory-based learning package TIMBL (Daelemans et al. 2002) and the rule induction package RIPPER (Cohen 1995). Independently of the type of data set, some clear tendencies could be observed with respect to precision and recall scores (Table 4.3). The precision scores for TIMBL were up to about 30% lower than the ones for RIPPER. For the recall scores, the opposite tendency could be observed, but to a lesser degree: TIMBL generally obtains a higher recall than RIPPER. Based on these tendencies we formulated some conclusions in terms of 'bias':

- With respect to the large difference in precision scores, we hypothesized that this was mainly due to the different feature handling in both learning techniques: RIPPER uses embedded feature selection for the construction of its rules, whereas TIMBL performs feature weighting, without taking into account the dependencies between features. One implication of this use of feature weighting is that a large group of features with low informativeness can overrule more informative features. This hypothesis was further investigated in Chapter 5.
- Furthermore, with respect to the lower recall scores for RIPPER, we hypothesized that the rule induction approach was more sensitive to the skewed class distribution in our data sets. In a lazy learning approach, all instances are stored in memory and no attempt is made to simplify the model by eliminating low frequency events, whereas in a eager learning approach such as RIPPER, possibly interesting information from the training data is either thrown away by pruning or made inaccessible by the eager construction of the model. For our data sets, this implies that RIPPER prunes away possibly interesting low-frequency positive data, which is harmful for its recall scores. This hypothesis was further investigated in Chapter 7.

With respect to the use of three classifiers trained on a different type of coreferential NPs, instead of one single classifier, we could observe that the RIPPER results of the three classifiers were always higher than the single classifier results, whereas the TIMBL results of the three classifiers were similar or even significantly below the single classifier results.

Feature selection Although the search for disambiguating features is central in the machine learning research for coreference resolution and for NLP tasks in general, the importance of feature selection has only recently been systematically investigated, as in Soon et al. (2001) and Ng and Cardie (2002c). In our experiments reported in Chapter 5, we opted for a more systematic and verifiable feature selection approach. We used three automated techniques for the selection of the relevant features, viz. backward elimination, bidirectional hillclimbing and a genetic algorithm. These three approaches start the search at a different starting point, when searching the space of feature subsets. The main objective was to determine the effect of feature selection on classifier performance. For TIMBL, we hypothesized that feature selection would lead to an increase of the precision scores. Feature selection indeed lifted the precision scores for TIMBL with up to 35% (Table 5.4, 5.5). As expected, this increase was much smaller (always less than 4%) for RIPPER due to the embedded feature selection used for the construction of the rules.

In these experiments, we also investigated whether the information sources considered to be optimal for one learner could be generalized to the other learner. With respect to the selected features, we observed that no general conclusions could be drawn (e.g. Table 5.6). Per language, per NP type data set (pronouns, proper nouns and common nouns) and per selection procedure, a different feature combination was selected by each learning algorithm. We concluded that the optimal feature selection had to be determined experimentally for each single data set. We consider this a rather disappointing result since this implies that the importance of the information sources cannot be considered as an isolated phenomenon. We were not able to determine a global set of features which holds for the task of coreference resolution. The whole experimental context with factors such as algorithm bias, algorithm parameters (Chapter 5) and class distribution (Chapter 7) interacts with the selection of information sources.

Parameter optimization In Chapter 5, we investigated the effect of parameter optimization on classifier performance. The main motivation for these experiments was that although most learning systems provide sensible default settings, it is by no means certain that they will be *optimal* parameter settings for some particular task. We performed an exhaustive variation of a number

of TIMBL and RIPPER parameters. Although the badly performing parameter combinations were in the minority, all experiments (Table 5.2, 5.3) revealed a lot of variation in the $F_{\beta=1}$ results when varying the algorithm parameters. We observed that the method-internal performance differences could be much larger than the method-comparing performance differences. For both learners we could conclude that parameter optimization overall leads to large performance increases (Table 5.7, 5.8).

In the parameter optimization experiments, we again investigated whether the optimal parameters for a given learning method could be generalized to the different data sets. However, no general conclusion could be drawn concerning these settings (Table 5.7, 5.8). The optimal settings merely revealed some tendencies, such as the predominant use of MVDM (Modified Value Difference Metric) and weighted voting for TIMBL, and the above average use of minimal description length instance ordering and a below zero loss ratio value for RIPPER. This predominant use of MVDM has also been observed in the experiments investigating the effect of the interaction of feature selection and parameter optimization. In fact, through the use of MVDM, a combination of supervised and unsupervised learning is obtained. This metric can be considered as a clustering approach in which similar feature values are grouped in clusters which are relevant for the task. Also for other NLP tasks (Buchholz 2002), this metric has been shown to perform well.

In a next optimization step, we investigated if the above described information sources and algorithm parameters also interact. These experiments were conducted since there appears to be little understanding in the current literature of the interaction between these variables. In case optimization is performed, this is mostly done sequentially, which may not be not advisable if different experimental factors interact.

Interaction of feature selection and parameter optimization In Chapter 6, we investigated the effect of the interaction of feature selection and parameter optimization. We used a genetic algorithm as a feasible method to do this costly optimization. The GA optimization experiments confirm the tendencies observed in the isolated feature selection and parameter optimization experiments (Table 6.1):

Feature selection, parameter optimization and their joint optimization can cause large variation in the results of both classifiers. All three optimization steps lead to a large improvement over the default results. Furthermore, optimization mainly wipes out the initial weaknesses of TIMBL and RIPPER in their default settings: the increase of $F_{\beta=1}$ scores for TIMBL is mainly obtained through a large increase of precision scores for TIMBL, whereas the increase of $F_{\beta=1}$ scores

for RIPPER is mainly due to the increase of recall scores. Furthermore, we could once again observe that the performance differences inside one single learning method could be much larger than the method-comparing performance differences. Also, the optimization results did not reveal a clear supremacy of one learner over the other, which once again confirms the necessity of optimization.

With respect to the use of three classifiers, each trained on the coreferential relations of a specific type of NP, instead of one single classifier covering all coreferential relations, the following could be observed. Three classifiers, each trained on one specific NP type, perform better than one single classifier in 5 out of 6 data sets. But since this difference in performance is only significant in half of the cases, we concluded that no convincing evidence was found for our initial hypothesis that three more specialized classifiers, each trained on the coreferential relations of a specific type of NP would perform better on the task of coreference resolution than one single classifier covering all coreferential relations.

We also investigated whether general conclusions could be drawn with respect to the selected features and optimal parameters.

- The following observations could be made with respect to the selected features: RIPPER selects fewer features than TIMBL, which can be explained through the different feature handling in both learning techniques. For RIPPER, a feature is either on or off. For TIMBL, a feature is either on, off or MVDM and it also incorporates different feature weighting techniques to assign different degrees of informativeness to the selected features.

  Furthermore, with respect to the informativeness of the features, we could observe that all features are informative for our task of coreference resolution. This observation refines the results displayed in Table 5.1 and also those reported by Soon et al. (2001), which show the lack of informativeness of the majority of the features, when they are considered in isolation. Furthermore, we could also observe that the initial predominance of the string-matching features (as also observed by Soon et al. (2001), Yang et al. (2004b) and others) has disappeared in favour of a more balanced combination of features.

  A last observation made with respect to the selected features, was that the feature selection considered to be optimal for TIMBL could be different from the one optimal for RIPPER. TIMBL and RIPPER often incorporate different features in their instances (see for example Figure 6.3).

- Although the parameter settings which were selected after optimization could not be generalized, not even within one single data set and although the parameter settings that were optimal when using all features were

not necessarily optimal when performing feature selection, some general observations could be made.

For TIMBL, we could see that 99% of all optimal individuals consisted of a combination of features for which the distance calculation is handled by the overlap metric and features handled by the MVDM metric. With respect to the *distance weights*, we could observe that the different distance weighted class voting schemes were preferred above the default majority voting (9%). Furthermore, 97% of the different selected values of k was higher than the default k=1, which could be explained trough the use of the MVDM metric in nearly all optimal individuals.[1]

For RIPPER, the most noticeable observation was made with respect to the *loss ratio* parameter, which allows to change the ratio of the cost of a false negative to the cost of a false positive. The default value of 1 was selected in only 3% of the cases, whereas all other individuals had a loss ratio value below 1 which implies that more importance is given to an improvement of the recall. This focus of recall can be explained through the skewedness of the data and the sensitivity of RIPPER to this skewedness (as investigated in Chapter 7). Since the positive class only represents a small fraction of the data, a large number of errors is made on the positive minority class. By decreasing the loss ratio value, an improvement on the recall scores can be obtained. Another clear observation was that the ordering method in which the classes are ordered by increasing frequency was selected in two thirds of the individuals (78%), whereas the ordering method which orders the classes by decreasing frequency was never selected. This parameter selection choice can again be explained through the skewed class distribution. For such a data set, an ordering method in which the classes are ordered by increasing frequency, makes more sense. This implies that first rules are learned for the positive minority class, whereas the negative class is taken as default classification. With respect to the number of *optimization passes* taken over the rules RIPPER learns, the default value 2 was selected in 91% of the individuals.

Class distribution In Chapter 7, we investigated how the class distribution of the data affects learning. In order to investigate the effect of class distribution on the performance of TIMBL and RIPPER, we created a variety of class distributions through the use of down-sampling and by changing the loss ratio parameter in RIPPER. For the down-sampling experiments we could conclude for the two learning methods that a decreasing rate of negative instances was beneficial for

[1] The MVDM metric groups feature values by looking at co-occurrence of values with target classes; this implies that the nearest neighbour set will usually be much smaller for MVDM than for the overlap metric at the same value of k.

recall. The same conclusion could be drawn in the experiments in which the loss ratio parameter was varied for RIPPER. Another general conclusion was that both down-sampling and a change of the loss ratio parameter below 1 was harmful for precision. We also showed that both learning approaches behave quite differently in case of skewedness of the classes and that they also react differently to a change in class distribution. TIMBL, which performs better on the minority class than RIPPER in case of a largely imbalanced class distribution, mainly suffers from a rebalancing of the data set. In contrast, the RIPPER results are sensitive to a change of class distribution or loss ratio. A decrease of the number of negative instances counters this pruning.

All these observations, however, are not limited to the task of coreference resolution. In earlier work (Hoste et al. 2002, Daelemans and Hoste 2002, Daelemans, Hoste, De Meulder and Naudts 2003, Decadt et al. 2004), we came to similar conclusions for the task of word sense disambiguation, the prediction of diminutive suffixes and part-of-speech tagging and for some non-NLP data sets.

In a typical comparative machine learning of language experiment, the impact of the factors discussed here is too often underestimated. In most comparative machine learning experiments, at least in computational linguistics, two or more algorithms are compared for a fixed sample selection, feature selection, feature representation, and (default) algorithm parameter setting over a number of trials (cross-validation), and if the measured differences are statistically significant, conclusions are drawn about which algorithm is better suited and why (mostly in terms of algorithm bias). Sometimes different sample sizes are used to provide a learning curve, and sometimes a limited parameter optimization is performed. No overall optimization of parameters, architecture and feature representation is undertaken (e.g. Mooney (1996), Escudero et al. (2000), Ng and Lee (1996), Lee and Ng (2002)). These studies explore only a few points in the space of possible experiments for each algorithm to be compared.

This methodology has already been criticized by Banko and Brill (2001), who showed that increasing the data sample size can strongly affect comparative results. In this study, we showed that changing any of the architectural variables, such as algorithm parameters, information sources and class distribution, can have great effects on the performance of a learning method, making questionable 'hard' conclusions in the literature on the relative adequacy of machine learning methods for a given task or on the importance of the information sources for solving a task, based on default settings of algorithms or on limited optimization only. Our experiments showed that there is a high risk that other areas in the experimental space may lead to radically different results and conclusions. In general, we conclude that the more effort is put in optimization, through feature selection, parameter optimization, sample selection and their joint optimization, the more reliable the results and the comparison will be.

9.2 Future research goals

Our future research goals relate to the observations made in Chapter 8. In this chapter, we showed that the results obtained on the MUC-6 (TIMBL: 64.3% and RIPPER: 63.4%) and MUC-7 (TIMBL: 60.2% and RIPPER: 57.6%) data sets were comparable to the results reported by Soon et al. (2001). Although the best results reported to date on the MUC-6 and MUC-7 data (Ng and Cardie 2002a, Ng and Cardie 2002b, Ng and Cardie 2002c) are significantly higher (69.5% on MUC-6 and 63.4% F_β on MUC-7), the field of coreference resolution still presents some major challenges. Furthermore, the F_β score of 51% of both TIMBL and RIPPER on the Dutch data showed that coreference resolution for Dutch is even more challenging.

Although we cannot quantify the error load of the different types of errors, since this would require a complete analysis of the different test corpora, we could get an impression of the major sources of errors through a qualitative error analysis of three English and three Dutch texts. We will now discuss the observations made for the different types of NPs and discuss some directions for future research.

With respect to *pronominal coreference*, we observed for both languages that a large part of the errors made by the pronominal resolution system involves the false distinction between anaphoric and pleonastic pronouns. Therefore, more effort should be put in features which can capture this difference. Another possible approach is to train a classifier, as in Mitkov et al. (2002), which automatically classifies instances of "it" as pleonastic or nominal anaphora.

For Dutch, the resolution of the pronominal anaphors is also severely hindered by part-of-speech tagging errors (e.g. the female "ze" is often erroneously tagged as a third person plural noun and vice versa). Since preprocessing errors are also a major source of errors for the Dutch proper noun and common noun resolution, we must conclude that the shallow parser trained on the Spoken Dutch Corpus is not suitable for this type of corpus. Therefore, we conclude that the whole part-of-speech tagging, NP chunking and relation finding procedure for Dutch should be reconsidered.

Furthermore, for the Dutch male and female pronouns, such as "hij", "hem", "zijn", "haar", we saw that the search space of candidate antecedents is much larger than that for the corresponding English pronouns, since they can also refer to the linguistic gender of their antecedent. The current feature vectors describing this type of relation have a low informativeness. Therefore, for the resolution of anaphors referring to the linguistic gender of their antecedent other features should be considered.

With respect to *proper noun coreference*, high precision scores ranging between 78.0% and 83.0% could be observed over all data sets. For both Dutch and English, the errors on the proper nouns are mainly caused by preprocessing errors: errors in NP chunking, part-of-speech tagging, relation finding, named entity recognition, apposition detection and alias detection. Therefore, more attention should be given to each of these preprocessing steps.

With respect to *common noun coreference*, we could observe low precision scores ranging between 57.2% and 47.5% on the three data sets. A similar observation was made by Ng and Cardie (2002a) and Strube et al. (2002) (for German). As for coreference resolution of anaphoric proper nouns, errors were caused by preprocessing errors, such as part-of-speech tagging, NP chunking, apposition recognition, etc. Other errors, such as the lack of detecting synonyms, hypernyms and paraphrases, are typical for the resolution of coreferential relations between common nouns. For this type of coreferential relations a large amount of semantic and world knowledge is required. In the construction of the instances for the English and Dutch data, we used WordNet and the Dutch EuroWordNet to build a set of semantic features. But both lexical resources, and in particular the Dutch EuroWordNet are restricted and miss a lot of commonly used expressions and their lexical relations. Furthermore, a lot of coreferential relations between NPs are restricted in time, such as the pair "Chirac"-"the president of France", or names of political parties (e.g. "VLD"-"de Vlaamse liberalen", "Agalev"- "de groenen"). In order to overcome the lack of information in the existing resources and in order to capture "dynamic" coreferential relations, we plan to use the Web as a resource (as for example Keller, Lapata and Ourioupina (2002), Turney (2001), Modjeska, Markert and Nissim (2003), and Bunescu (2003)).

Bibliography

Aha, D.: 1997, Lazy learning: Special issue editorial, *Artificial Intelligence Review* **11**, 7–11.

Aha, D. and Bankert, R.: 1996, A comparative evaluation of sequential feature selection algorithms, *in* D. Fischer and J.-H. Lenz (eds), *Artificial intelligence and statistics V*, New York: Springer Verlag.

Aha, D., Kibler, D. and Albert, M.: 1991, Instance-based learning algorithms, *Machine Learning* **6**, 37–66.

Aone, C. and Bennett, S.: 1995, Evaluating automated and manual acquisition of anaphora resolution strategies, *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL-1995)*, pp. 122–129.

Baayen, R., Piepenbrock, R. and van Rijn, H.: 1993, The celex lexical data base on cd-rom.

Baldwin, B.: 1997, Cogniac: high precision coreference with limited knowledge and linguistic resources, *Proceedings of the ACL'97/EACL'97 workshop on Operational Factors in Practical, Robust Anaphora Resolution*, pp. 38–45.

Baldwin, B., Morton, T., Bagga, A., Baldridge, J., Chandraseker, R., Dimitriadis, A., Snyder, K. and Wolska, M.: 1998, Description of the upenn camp system as used for coreference, *Proceedings of the Seventh Message Understanding Conference (MUC-7)*.

Banko, M. and Brill, E.: 2001, Scaling to very very large corpora for natural language disambiguation, *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL-01)*, pp. 26–33.

Blake, C. and Merz, C.: 2000, Uci repository of machine learning databases, Department of Information and Computer Science, University of California at Irvine, CA. http://www.ics.uci.edu/˜mlearn/MLrepository.html.

Blum, A. and Langley, P.: 1997, Selection of relevant features and examples in machine learning, *Artificial Intelligence* **97**(1-2), 245–271.

Blum, A. and Mitchell, T.: 1998, Combining labeled and unlabeled data with co-training, *Proceedings of the Workshop on Computational Learning Theory*, pp. 92–100.

Bouma, G.: 2003, Doing dutch pronouns automatically in optimality theory, *Proceedings of the EACL 2003 Workshop on The Computational Treatment of Anaphora*.

Breiman, L.: 1996, Bagging predictors, *Machine Learning* **24**, 123–140.

Brennan, S., Friedman, M. and Pollard, C.: 1987, A centering approach to pronouns, *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics (ACL-87)*, pp. 155–162.

Buchholz, S.: 2002, *Memory-based Grammatical Relation finding*, PhD thesis, Tilburg University.

Buchholz, S., Veenstra, J. and Daelemans, W.: 1999, Cascaded grammatical relation assignment, *Proceedings of EMNLP/VLC-99*, pp. 239–246.

Bunescu, R.: 2003, Associative anaphora resolution: A web-based approach, *Proceedings of the EACL 2003 Workshop on the Computational Treatment of Anaphora*, pp. 47–52.

Byron, D. and Allen, J.: 1999, Applying genetic algorithms to pronoun resolution, *Proceedings of the Sixteenth National Conference on Artificial Intelligence*.

Carbonell, J. and Brown, R.: 1988, Anaphora resolution: a multi-strategy approach, *Proceedings of the 12th International Conference on Computational Linguistics (COLING-1988)*, pp. 96–101.

Cardie, C. and Howe, N.: 1997, Improving minority class prediction using case-specific feature weights, *Proceedings of the 14th International Conference on Machine Learning (ICML-1997)*, pp. 57–65.

Cardie, C. and Wagstaff, K.: 1999, Noun phrase coreference as clustering, *Proceedings of the 1999 joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pp. 82–89.

Carter, D.: 1987, *Interpreting Anaphors in Natural Language Texts*, Ellis Horwood, Chichester, U.K.

Caruana, R. and Freitag, D.: 1994, Greedy attribute selection, *Proceedings of the International Conference on Machine Learning (ICML-1994)*, pp. 28–36.

Chan, P. and Stolfo, S.: 1998, Toward scalable learning with non-uniform class and cost distributions: A case study in credit card fraud detection, *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*, pp. 164–168.

Chawla, N., Bowyer, K., Hall, L. and Kegelmeyer, W.: 2002, Smote: Synthetic minority over-sampling technique, *Journal of Artificial Intelligence Research (JAIR)* **16**, 321–357.

Clark, H.: 1975, Bridging, *Proceedings of the Conference on Theoretical Issues in NLP*, pp. 169–174.

Cohen, W. W.: 1995, Fast effective rule induction, *Proceedings of the 12th International Conference on Machine Learning (ICML-1995)*, pp. 115–123.

Connolly, D., Burger, J. and Day, D.: 1994, A machine learning approach to anaphoric reference, *Proceedings of the International Conference on 'New Methods in Language Processing'*.

Cooper, R.: 1979, The interpretation of pronouns, *Syntax and Semantics* **10**, 61–93.

Cost, S. and Salzberg, S.: 1993, A weighted nearest neighbour algorithm for learning with symbolic features, *Machine Learning* **10**, 57–78.

Cover, T. and Hart, P.: 1967, Nearest neighbor pattern classification, *Institute of Electrical and Electronics Engineers Transactions on Information Theory* **13**, 21–27.

Crow, J. and Kimura, M.: 1970, *An Introduction to Population Genetics Theory*, New York: Harper and Row.

Daelemans, W. and Hoste, V.: 2002, Evaluation of machine learning methods for natural language processing tasks, *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC-2002)*, pp. 755–760.

Daelemans, W., Hoste, V., De Meulder, F. and Naudts, B.: 2003, Combined optimization of feature selection and algorithm parameter interaction in machine learning of language, *Proceedings of the 14th European Conference on Machine Learning (ECML-2003)*, pp. 84–95.

Daelemans, W., van den Bosch, A. and Weijters, A.: 1997, Igtree: using trees for compression and classification in lazy learning algorithms, *Artificial Intelligence Review* **11**, 407–423.

Daelemans, W., van den Bosch, A. and Zavrel, J.: 1999, Forgetting exceptions is harmful in language learning, *Machine Learning* **34**(1-3), 11–41.

Daelemans, W., Zavrel, J., Berck, P. and Gillis, S.: 1996, Mbt: A memory-based part of speech tagger generator, *Proceedings of the 4th ACL/SIGDAT Workshop on Very Large Corpora*, pp. 14–27.

Daelemans, W., Zavrel, J., van den Bosch, A. and van der Sloot, K.: 2003, Memory based tagger, version 2.0, reference guide, *Technical Report ILK Technical Report - ILK 03-13*, Tilburg University.

Daelemans, W., Zavrel, J., van der Sloot, K. and van den Bosch, A.: 2002, Timbl: Tilburg memory-based learner, version 4.3, reference guide, *Technical Report ILK Technical Report - ILK 02-10*, Tilburg University.

Dagan, I. and Itai, A.: 1990, Automatic processing of large corpora for the resolution of anaphora references, *Proceedings of the 13th International Conference on Computational Linguistics (COLING-1990)*, pp. 330–332.

Dagan, I., Justeson, J., Lappin, S., Leass, H. and Ribak, A.: 1995, Syntax and lexical statistics in anaphora, *Applied Artificial Intelligence* **9**(6), 633–644.

Davies, S., Poesio, M., Bruneseaux, F. and Romary, L.: 1998, Annotating coreference in dialogues: Proposal for a scheme for mate. http://www.hcrc.ed.ac.uk/ poesio/MATE/anno$_m$anual.htm.

De Pauw, G., Laureys, T., Daelemans, W. and Van hamme, H.: 2004, A comparison of two different approaches to morphological analysis of dutch, *Proceedings of the Seventh Meeting of the ACL Special Interest Group in Computational Phonology*, pp. 62–69.

Decadt, B., Hoste, V., Daelemans, W. and van den Bosch, A.: 2004, Gambl, genetic algorithm optimization of memory-based wsd, *Proceedings of the Third International Workshop on the Evaluation of Systems for Semantic Analysis of Text (SENSEVAL-3)*, pp. 108–112.

Demeulder, F. and Daelemans, W.: 2003, Memory-based named entity recognition using unannotated data, *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL-2003)*, pp. 208–211.

Domingos, P.: 1999, Metacost: A general method for making classifiers cost sensitive, *Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining*, pp. 155–164.

Drummond, C. and Holte, R.: 2003, C4.5, class imbalance, and cost sensitivity: Why under-sampling beats over-sampling, *Proceedings of the Workshop on Learning from Imbalanced Datasets II*.

Dudani, S.: 1976, The distance-weighted k-nearest-neighbor rule, *IEEE Transactions on Systems, Man and Cybernetics* **6**(4), 325–327.

Escudero, G., Marquez, L. and Rigau, G.: 2000, Boosting applied to word sense disambiguation, *European Conference on Machine Learning*, pp. 129–141.

Fan, W., Stolfo, S., Zhang, J. and Chan, P.: 1999, Adacost: Misclassification cost-sensitive boosting, *Proceedings of the Sixteenth International Conference on Machine Learning (ICML-1999)*, pp. 97–105.

Fawcett, T.: 2003, Roc graphs: Notes and practical considerations for researchers, *Technical Report Tech report HPL-2003-4*, HP Laboratories, Palo Alto, CA, USA.

Fellbaum, C.: 1998, *WordNet: An Electronic Lexical Database*, MIT Press.

Fisher, F., Soderland, S., Mccarthy, J., Feng, F. and Lehnert, W.: 1995, Description of the umass system as used for muc-6, *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pp. 127–140.

Fligelstone, S.: 1990, *A description of the conventions used in the Lancaster Anaphoric Treebank Scheme*, Department of Linguistics and Modern English Language, Lancaster University.

Fraurud, K.: 1992, *Processing Noun Phrases in Natural Discourse*, PhD thesis, Stockholm University.

Freund, Y. and Schapire, R.: 1996, Experiments with a new boosting algorithm, *Proceedings of the Thirteenth International Conference on Machine Learning (ICML-1996)*, pp. 148–156.

Furnkranz, J. and Widmer, G.: 1994, Incremental reduced error pruning, *Proceedings of the 11th International Conference on Machine Learning (ICML-1994)*, pp. 70–77.

Gardent, C.: 2000, Deaccenting and higher-order unification, *Journal of Logic, Language and Information* **9**(3), 313–338.

Ge, N., Hale, J. and Charniak, E.: 1998, A statistical approach to anaphora resolution, *Proceedings of the Sixth Workshop on very Large Corpora*, pp. 161–170.

Goldberg, D.: 1989, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley.

Goldberg, D. and Deb, K.: 1991, A comparative analysis of selection schemes used in genetic algorithms, *Foundations of Genetic Algorithms*, Morgan Kaufmann Publishers, San Mateo, California, USA, pp. 69–93.

Grosz, B., Joshi, A. and Weinstein, S.: 1983, Providing a unified account of definite noun phrases in discourse, *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics (ACL-83)*, pp. 44–50.

Grosz, B., Joshi, A. and Weinstein, S.: 1995, Centering: a framework for modeling the local coherence of discourse, *Computational Linguistics* **21**(2), 203–225.

Harabagiu, S., Bunescu, R. and Maiorano, S.: 2001, Text and knowledge mining for coreference resolution, *Proceedings of the 2nd Meeting of the North American Chapter of the Association of Computational Linguistics (NAACL-2001)*, pp. 55–62.

Hardt, D.: 1992, Vp ellipsis and contextual interpretation, *Proceedings of the International Conference on Computational Linguistics (COLING-92)*.

Hardt, D.: 2004, Dynamic centering, *Proceedings of the Workshop on Reference Resolution and its Applications*, pp. 55–62.

Hartrumpf, S.: 2001, Coreference resolution with syntactico-semantic rules and corpus statistics, *Proceedings of the Fifth Conference on Computational Natural Language Learning (CoNLL-2001)*, pp. 137–144.

Hawkins, J.: 1978, *Definiteness and Indefiniteness, A Study in Reference and Grammaticality Prediction*, Humanities Press, Atlantic Highlands, NJ.

Heim, I.: 1982, *The Semantics of Definite and Indefinite Noun Phrases*, PhD thesis, University of Massachusetts at Amherst.

Hemphill, C., Godfrey, J. and Doddington, G.: 1990, The atis spoken language system pilot corpus, *Proceedings of the DARPA Speech and Natural Language Workshop*, pp. 96–101.

Hirschman, L. and Chinchor, N.: 1998, Muc-7 coreference task definition. version 3.0, *Proceedings of the Seventh Message Understanding Conference (MUC-7)*.

Hirschman, L., Robinson, P., Burger, J. and Vilain, M.: 1997, Automating coreference: The role of annotated training data, *Proceedings of the AAAI Spring Symposium on Applying Machine Learning to Discourse Processing*.

Hirst, G.: 1981, Anaphora in natural language understanding: A survey, *Lecture Notes in Computer Science*, Vol. 119, Springer-Verlag Berlin Heidelberg New York.

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

Holland, J.: 1975, *Adaptation in natural and artificial Systems*, MIT Press.

Holte, R.: 1993, Very simple classification rules perform well on most commonly used datasets, *Machine Learning* **11**, 63–90.

Hoste, V., Hendrickx, I., Daelemans, W. and van den Bosch, A.: 2002, Parameter optimization for machine-learning of word sense disambiguation, *Natural Language Engineering, Special Issue on Word Sense Disambiguation Systems* **8**, 311–325.

Howe, N. and Cardie, C.: 1997, Examining locally varying weights for nearest neighbor algorithms, *Proceedings of the Second International Conference on Case-Based Reasoning*, pp. 455–466.

Japkowicz, N. and Stephen, S.: 2002, The class imbalance problem: A systematic study, *Intelligent Data Analysis Journal* **6**(5), 429–450.

John, G., Kohavi, R. and Pfleger, K.: 1994, Irrelevant features and the subset selection problem, *International Conference on Machine Learning*, pp. 121–129.

Joshi, M., Kumar, V. and Agarwal, R.: 2001, Evaluating boosting algorithms to classify rare classes: Comparison and improvements, *Proceedings of the First IEEE International Conference on Data Mining*, pp. 257–264.

Kamp, H.: 1981, A theory of truth and semantic representation, *in* J. Groenendijk, T. Janssen and M. Stokhof (eds), *Formal methods in the study of language*, Mathematical Centre, Amsterdam, pp. 277–322.

Karttunen, L.: 1976, Discourse referents, *Syntax and Semantics* **7**.

Kehler, A.: 1997, Probabilistic coreference in information extraction, *in* R. I. Providence (ed.), *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing (EMNLP-97)*.

Keller, F., Lapata, M. and Ourioupina, O.: 2002, Using the web to overcome data sparseness, *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*, pp. 230–237.

Kennedy, C. and Boguraev, B.: 1996, Anaphora for everyone: Pronominal anaphora resolution without a parser, *Proceedings of the 16th International Conference on Computational Linguistics (COLING-1996)*, pp. 113–118.

Kibble, R.: 2000, Coreference annotation: Whither?, *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC-2000)*, pp. 1281–1286.

Kohavi, R. and John, G. H.: 1997, Wrappers for feature subset selection, *Artificial Intelligence* **97**(1-2), 273–323.

Kolodner, J.: 1993, *Case-based reasoning*, Morgan Kaufmann, San Mateo, CA.

Kool, A., Daelemans, W. and Zavrel, J.: 2000, Genetic algorithms for feature relevance assignment in memory-based language processing, *Proceedings of the Fourth Conference on Computational Natural Language Learning and of the Second Learning Language in Logic Workshop*, pp. 103–106.

Kool, A., Zavrel, J. and Daelemans, W.: 2000, Simultaneous feature selection and parameter optimization for memory-based natural language processing, *Proceedings of the 10th BENELEARN meeting*, pp. 93–100.

Kubat, M., Holte, R. and Matwin, S.: 1997, Learning when negative examples abound, *Proceedings of the Ninth European Conference on Machine Learning (ECML-1997)*, pp. 146–153.

Kubat, M., Holte, R. and Matwin, S.: 1998, Machine learning for the detection of oil spills in satellite radar images, *Machine Learning* **30**, 195–215.

Kucera, H. and Francis, W.: 1967, *Computational analysis of present-day English*, Brown University Press, RI.

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

Lee, Y. and Ng, H.: 2002, An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation, *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*, pp. 41–48.

Levenhstein, V.: 1966, Binary codes capable of correcting deletions, insertions and reversals, *Soviet Physics Daklady* **10**, 707–710.

Lewis, D. and Gale, W.: 1994, Training text classifiers by uncertainty sampling, *Proceedings of the Seventh Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 3–12.

Luo, X., Ittycheriah, A., Jing, H., Kambhatla, N. and Roukos, S.: 2004, A mention-synchronous coreference resolution algorithm based on the bell tree, *in* S. Barcelona (ed.), *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-2004)*, pp. 136–143.

Maloof, M.: 2003, Learning when data sets are imbalanced and when costs are unequal and unknown, *Proceedings of the Workshop on Learning from Imbalanced Data Sets II*.

Manevitz, L. and Yousef, M.: 2001, One-class svms for document classification, *Journal of Machine Learning Research* **2**, 139–154.

Marcus, M. P., Santorini, B. and Marcinkiewicz, M. A.: 1993, Building a large annotated corpus of english: The penn treebank, *Computational Linguistics* **19**(2), 313–330.

McCarthy, J.: 1996, *A Trainable Approach to Coreference Resolution for Information Extraction*, PhD thesis, Department of Computer Science, University of Massachusetts, Amherst MA.

McCarthy, J. and Lehnert, W.: 1995, Using decision trees for coreference resolution, *Proceedings of the Fourteenth International Conference on Artificial Intelligence*, pp. 1050–1055.

Michalewicz, Z.: 1992, *Genetic algorithms + Data Structures = Evolution Programs*, Springer-Verlag.

Mihalcea, R.: 2002, Word sense disambiguation with pattern learning and automatic feature selection, *Natural Language Engineering, Special Issue on Word Sense Disambiguation Systems* **8**, 343–358.

Mitchell, M.: 1996, *An Introduction to Genetic Algorithms*, MIT Press.

Mitkov, R.: 1998, Robust pronoun resolution with limited knowledge, *Proceedings of the 17th International Conference on Computational Linguistics (COLING-1998/ACL-1998)*, pp. 869–875.

Mitkov, R.: 2002, *Anaphora Resolution*, Longman.

Mitkov, R., Evans, R. and Orasan, C.: 2002, A new, fully automatic version of mitkov's knowledge-poor pronoun resolution method, *Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2002)*.

Mitkov, R., Evans, R., Orasan, C., Barbu, C., Jones, L. and Sotirova, V.: 2000, Coreference and anaphora: developing annotating tools, annotated resources and annotation strategies, *Proceedings of the Discourse Anaphora and Anaphora Resolution Colloquium (DAARC-2000)*, pp. 49–58.

Modjeska, N., Markert, K. and Nissim, M.: 2003, Using the web in machine learning for other-anaphora resolution, *Proceedings of the 2003 Conference on Empirical Methods in Natural Lanugage Processing (EMNLP-2003)*, pp. 176–183.

Mooney, R.: 1996, Comparative experiments on disambiguating word senses: An illustration of the role of bias in machine learning, *in* E. Brill and K. Church (eds), *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 82–91.

MUC-6: 1995, Coreference task definition. version 2.3., *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pp. 335–344.

MUC-7: 1998, Muc-7 coreference task definition. version 3.0., *Proceedings of the Seventh Message Understanding Conference (MUC-7)*.

Müller, C., Rapp, S. and Strube, M.: 2002, Applying co-training to reference resolution, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002)*, pp. 352–359.

Ng, H. and Lee, H.: 1996, Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach, *Proceedings of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics (ACL-96)*, pp. 40–47.

Ng, V. and Cardie, C.: 2002a, Combining sample selection and error-driven pruning for machine learning of coreference rules, *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*, pp. 55–62.

Ng, V. and Cardie, C.: 2002b, Identifying anaphoric and non-anaphoric noun phrases to improve coreference resolution, *Proceedings of the 19th International Conference on Computational Linguistics (COLING-2002)*.

Ng, V. and Cardie, C.: 2002c, Improving machine learning approaches to coreference resolution, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002)*, pp. 104–111.

Ng, V. and Cardie, C.: 2003, Bootstrapping coreference classifiers with multiple learning algorithms, *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing (EMNLP-2003)*, pp. 113–120.

Noreen, E.: 1989, *Computer intensive methods for testing hypothesis: An introduction*, John Wiley & Sons, New York.

op den Akker, H., Hospers, M., Lie, D., Kroezen, E. and Nijholt, A.: 2002, A rule-based reference resolution method for dutch discourse, *Proceedings 2002 Symposium on Reference Resolution in Natural Language Processing*, pp. 59–66.

Orasan, C.: 2000, Clinka a coreferential links annotator, *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC-2000)*, pp. 491–496.

Orasan, C., Evans, R. and Mitkov, R.: 2000, Enhancing preference-based anaphora resolution with genetic algorithms, *Proceedings of NLP-2000*, pp. 185–195.

Palomar, M., Ferrández, A., Moreno, L., Martìnez-Barco, P., Peral, J., Saiz-Noeda, M. and Muñoz, R.: 2001, An algorithm for anaphora resolution in spanish texts, *Computational Linguistics* **27**(4), 545–567.

Partee, B.: 1973, Some structural analogies between tenses and pronouns in english, *Journal of Philosophy* **70**, 601–609.

Passoneau, R.: 1996, Instructions for applying discourse reference annotation for multiple applications (drama). Unpublished manuscript.

Passoneau, R. and Litman, D.: 1997, Discourse segmentation by human and automated means, *Computational Linguistics* **23**(1), 3–139.

Pazzani, M., Merz, C., Murphy, P., Ali, K., Hume, T. and Brunk, C.: 1994, Reducing misclassification costs, *Proceedings of the Eleventh International Conference on Machine Learning (ICML-1994)*, pp. 217–225.

Poesio, M., Stevenson, R., di Eugenio, B. and Hitzeman, J.: 2004, Centering: A parametric theory and its instantiations, *Computational Linguistics* **30**(3).

Poesio, M. and Vieira, R.: 1998, A corpus-based investigation of definite description use, *Computational Linguistics* **24**(2), 183–216.

Preiss, J.: 2002, Anaphora resolution with memory based learning, *Proceedings of CLUK-5*, pp. 1–9.

Quinlan, J.: 1993, *C4.5: Programs for machine learning*, Morgan Kaufmann, San Mateo, CA.

Quinlan, J.: 1995, Mdl and categorical theories (continued), *Proceedings of 12th International Conference on Machine Learning (ICML-1995)*, pp. 464–470.

Quinlan, J.: 1996, Boosting first-order learning, *Algorithmic Learning Theory, 7th International Workshop*, Sydney, Australia, pp. 143–155.

Raskutti, B. and Kowalczyk, A.: 2003, Extreme re-balancing for svms: a case study, *Proceedings of the Workshop on Learning from Imbalanced Datasets II*.

Rich, E. and LuperFoy, S.: 1988, An architecture for anaphora resolution, *Proceedings of the Second Conference on Applied Natural Language Processing*, pp. 18–24.

Riesbeck, C. and Schank, R.: 1989, *Inside Case-Based Reasoning*, Lawrence Erlbaum Associates, Cambridge, MA.

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

Skalak, D. B.: 1993, Using a genetic algorithm to learn prototypes for case retrieval and classification, *Proceedings of the AAAI-93 Case-Based Reasoning Workshop*, pp. 64–69.

Skalak, D. B.: 1994, Prototype and feature selection by sampling and random mutation hill climbing algorithms, *Proceedings of the Eleventh International Conference on Machine Learning*, pp. 293–301.

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

Stanfill, C. and Waltz, D.: 1986, Toward memory-based reasoning, *Communications of the ACM* **29**(12), 1213–1228.

Strube, M. and Hahn, U.: 1999, Functional centering–grounding referential coherence in information structure, *Computational Linguistics* **25**(3), 309–344.

Strube, M. and Müller, C.: 2003, A machine learning approach to pronoun resolution in spoken dialogue, *Proceedings of the 41th Annual Meeting of the Association for Computational Linguistics (ACL-2003)*, pp. 168–175.

Strube, M., Rapp, S. and Müller, C.: 2002, The influence of minimum edit distance on reference resolution, *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*, pp. 312–319.

Stuckardt, R.: 2001, Design and enhanced evaluation of a robust anaphor resolution algorithm, *Computational Linguistics* **27**(4), 473–506.

Tax, D.: 2001, *One-class classification*, PhD thesis, TU Delft.

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

Ting, K.: 2000, A comparative study of cost-sensitive boosting algorithms, *Proceedings of the Seventeenth International Conference on Machine Learning (ICML-2000)*, pp. 983–990.

Tjong Kim Sang, E., Daelemans, W. and Höthker, A.: 2004, Reduction of dutch sentences for automatic subtitling, *Computational Linguistics in the Netherlands 2003. Selected Papers from the Fourteenth CLIN Meeting*, pp. 109–123.

Trask, R.: 1983, *A Dictionary of Grammatical Terms in English*, Routledge, London and New York.

Turney, P.: 2001, Mining the web for synonyms: Pmi-ir versus lsa on toefl, *Proceedings of the 12th European Conference on Machine Learning*, pp. 491–502.

Tutin, A., Trouilleux, F., Clouzot, C., Gaussier, E., Zaenen, A., Rayot, S. and Antoniadis, G.: 2000, Annotating a large corpus with anaphoric links, *Proceedings of the Discource, Anaphora and Reference Resolution Conference (DAARC-2000)*, pp. 28–38.

van Deemter, K. and Kibble, R.: 2000, On coreferring: Coreference in muc and related annotation schemes, *Computational Linguistics* **26**(4), 629–637.

van Rijsbergen, C.: 1979, *Information Retrieval*, Buttersworth, London.

Veenstra, J., van den Bosch, A., Buchholz, S., Daelemans, W. and Zavrel, J.: 2000, Memory-based word sense disambiguation, *Computers and the Humanities* **34**(1/2), 171–177.

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

Vilain, M., Burger, J., Aberdeen, J., Connolly, D. and Hirschman, L.: 1995, A model-theoretic coreference scoring scheme, *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pp. 45–52.

Wagner, R. and Fisher, M.: 1974, The string-to-string correction problem, *Journal of ACM* **21**(1), 168–173.

Walker, M., Joshi, A. and Prince, E. e.: 1998, *Centering in Discourse*, Oxford University Press.

Webber, B.: 1978, *A Formal Approach to Discourse Anaphora*, PhD thesis, Harvard University.

Webber, B.: 1998, Tense as discourse anaphor, *Computational Linguistics* **14**(2), 61–73.

Weiss, G.: 2003, *The Effect of Small Disjuncts and Class Distribution on Decision Tree Learning*, PhD thesis, Department of Computer Science, Rutgers University, New Brunswick, New Jersey.

Weiss, S. M. and Kulikowski, C. A.: 1991, *Computer Systems That Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems*, Morgan Kaufmann, San Mateo, California.

White, A. and Liu, W.: 1994, Bias in information-based measures in decision tree induction, *Machine Learning* **15**(3), 321–329.

Wolpert, D. and Macready, W.: 1995, No free lunch theorems for search, *Technical Report SFI-TR-95-02-010*, Santa Fe Institute, Santa Fe, NM.

Yang, X., Su, S., Zhou, G. and Tan, C.: 2004a, Improving pronoun resolution by incorporating coreferential information of candidates, *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, Barcelona, Spain, pp. 128–135.

Yang, X., Su, S., Zhou, G. and Tan, C.: 2004b, A np-cluster approach to coreference resolution, *Proceedings of the 20th International Conference on Computational Linguistics (COLING-2004)*, Geneva, Switzerland.

Yang, X., Zhou, G., Su, S. and Tan, C.: 2003, Coreference resolution using competition learning approach, *Proceedings of the 41th Annual Meeting of the Association for Compuatiational Linguistics (ACL-03)*, Sapporo, Japan, pp. 176–183.

Yeh, A.: 2000, More accurate tests for the statistical significance of result differences, *Proceedings of the 18th International Conference on Computational Linguistics (COLING-2000)*, Saarbruecken, Germany, pp. 947–953.

Zhang, J.: 1992, Selecting typical instances in instance-based learning, *Proceedings of the International Machine Learning Conference*, pp. 470–479.

Zhang, J. and Mani, I.: 2003, knn approach to unbalanced data distributions: A case study involving information extraction, *Proceedings of the Workshop of Learning from Imbalanced Datasets II*.

Zhang, J., Mani, I., Lawrence, S., Burns, I., Back, A., Tsoi, A. and Giles, C.: 1998, Neural network classification and unequal prior class probabilities, *Tricks of the Trade, Lecture Notes in Computer Science State-of-the-Art Surveys*, Springer Verlag, pp. 299–314.

APPENDIX A

Manual for the annotation of coreferences in Dutch newspaper texts

A.1 Introduction

In many texts, the subject discussed is mentioned in different ways, for example:

> Na het ontslag van **Bert Anciaux** uit **de Vlaamse regering**, moest **minister-president Patrick Dewael** op zoek naar een vervanger voor **de minister van Cultuur**. **De minister van Spirit** diende **zijn** ontslag in bij **de minister-president** omdat **hij** de aantijgingen beu was. **Patrick Dewael** stelde **een Brusselse VLD'er** aan als **nieuwe minister in de Vlaamse regering**.

> English: After Bert Anciaux resigned from the Flemish government, prime minister Patrick Dewael had to look for a replacement for the Secretary of Cultural Affairs. The minister of the Spirit party tendered his resignation to the prime minister because he was sick of the allegations. Patrick Dewael appointed a VLD member from Brussels as the new minister in the Flemish government.

Subsequent mentions of a given entity can be the same (e.g. 2 times "de Vlaamse regering") or partially the same, as "minister-president Patrick Dewael", "de minister-president" and "Patrick Dewael" in the previous example. The same entity, however, can also be denoted by a different word/set of words: "Bert Anciaux" , "de minister van Cultuur", "De minister van Spirit" and the pronouns "zijn" and "hij" all refer to the same person. This type of expressions are also called **anaphoric or coreferential expressions**. The example clearly shows that a lot of elements in a text are correlated and these correlations have to be well understood in order to have a good text understanding.

In this manual, we will present some guidelines for the annotation of coreferences or anaphora in Dutch newspaper texts. We will first discuss some relevant terms, such as "coreference", "anaphora", "cospecification", etc. We will continue with a short description of some existing annotation schemes and we also give a brief introduction to the corpus to be annotated.

A.1.1 Coreference and anaphora

In this section, we will present some definitions for anaphora and coreference. In many work, "coreference" and "anaphora" are used interchangeably, e.g. Hirst (1981). He gives the following definition:

> ANAPHORA is the device of making in discourse an ABBREVIATED reference to some entity (or entities) in the expectation that the perceiver of the discourse will be able to disabbreviate the reference and thereby determine the identity of the entity. The reference is called an ANAPHOR, and the entity to which it refers is its REFERENT or ANTECEDENT. A reference and its referent are said to be COREFERENTIAL. The process of determining the referent of an anaphor is called RESOLUTION. By ABBREVIATED, I mean containing fewer bits of disambiguating information (in the sense of Shannon and Weaver 1949), rather than lexically or phonetically shorter. (p4-5)

The definition used by Hirschman and Chinchor (1998), Davies et al. (1998), Mitkov (2002) and many others is that two strings corefer when they point to the same entity in the world. In their annotation schemes, they do not make a difference between the terms "coreference" and "anaphora". This approach, however, has been criticized by Kibble (2000) and van Deemter and Kibble (2000). They claim that the coreferential annotations provided by, for example Hirschman and Chinchor (1998), go well beyond the annotation of the relation

of coreference proper. They give two textbook definitions from Trask (1983) to give a clear view on the differences between coreferences and anaphors:

- **Coreference**: The relation which obtains between two NPs (e.g. NP_1 and NP_2) both of which are interpreted as referring to the same extralinguistic entity (**Referent(NP)**). In short:

  NP_1 and NP_2 corefer if and only if Referent(NP_1) = Referent(NP_2).

  E.g. *Bert Anciaux nam vandaag ontslag uit de Vlaamse regering. De nummer 1 van SPIRIT was de kritiek moe. (English: Bert Anciaux resigned today from the Flemish government. The number 1 of SPIRIT was fed up with the criticism.)*
  Coreference pair: Bert Anciaux, de nummer 1 van SPIRIT.

  Features of coreference:

  - In the case of coreference, there is an *equivalence relation*: "Bert Anciaux" and "de nummer 1 van SPIRIT" refer to the same person.
  - Coreferential relations have two important features: they are *symmetrical* (if NP_1 and NP_2 corefer, this implies that also NP_2 and NP_1 corefer).
  - They are also *transitive* (if NP_1 and NP_2 corefer and if also NP_2 and NP_3 corefer, this implies that also NP_1 and NP_3 will corefer). This transitivity can alleviate the task of coreference resolution (as suggested in (McCarthy 1996)). But transitivity also implies that wrongly assigned coreference relations will cause even more errors.
  - Another feature from coreference relations is that there is *no context-sensitivity of interpretation*. "Bert Anciaux" and "de nummer 1 van SPIRIT" can corefer and they do not depend on each other for their interpretation.

- **Anaphor**: An item (e.g. NP_1) with little or no intrinsic meaning or reference which takes its interpretation from another item (e.g. NP_2) in the same sentence or discourse, its antecedent.

  NP_1 takes NP_2 as its anaphoric antecedent if and only if NP_1 depends on NP_2 for its interpretation.

  E.g. *Bert Anciaux zei dat hij de aantijgingen beu was. (English: Bert Anciaux said he was sick of the allegations.*
  Anaphor: hij; antecedent: Bert Anciaux

  Features of anaphors:

- An anaphoric relation is *nonsymmetrical*: if NP_1 is anaphoric to NP_2, then NP_2 is not necessarily anaphoric to NP_1.
- It is *nontransitive*.
- And it also implies *context-sensitivity of interpretation*, e.g. "hij" cannot be interpreted without its antecedent "Bert Anciaux".

Anaphoric and coreferential relations can coincide, but not all coreferential relations are anaphoric and not all anaphoric relations are coreferential. In the remainder of this manual, we will use the terms "coreference" and "anaphora" as synonyms, just as in the MUC-6 and MUC-7 manuals.

Another commonly used term is "**cospecification**". Following the definition of (Sidner 1979), an anaphoric expression and its antecedent cospecify when they denote the same object. Mostly, cospecification and coreference are identical. But sometimes there is cospecification without referring to any object in the real world, e.g. "De koning van Belgie heeft een buitenechtelijk kind. Hij heeft dit onlangs bevestigd. (English: The king has an illegitimate child. He confirmed this recently.)" "Hij" cospecifies with "De koning van Belgie", but it refers to an entity in discourse and not to an object in the real world. However, this definition of cospecification is not very clear and not very useful for annotation (see also van Deemter and Kibble (2000)).

A.1.2 Types of coreference

Annotating corpora with information about anaphoric/coreferential relations between elements of a text is useful both from a linguistic point of view and for applications such as information extraction. Because of the high frequency of anaphoric expressions, resolving them is important for text understanding.

There are many different types of coreferential relations (see for example Webber (1978), McCarthy (1996)), e.g.

- **identity relations** as in *Xavier Malisse heeft zich geplaatst voor de halve finale in Wimbledon. De Vlaamse tennisser zal dan tennissen tegen een onbekende tegenstander. (English: Xavier Malisse qualified for the semi-finals in Wimbledon. The Flemish tennis player will play against an unknown opponent.)* In the previous example, there is an identity relation between "Xavier Malisse" and "De Vlaamse tennisser".
- **type/token relations** as in *Ik verkies de rode auto, maar mijn man wou de grijze. (English: I prefer the red car, but my husband wanted the*

gray one.). In the example sentence, we are talking about two distinct cars, a red one and a gray one. "de grijze" denotes something like an object type rather than an object token. Also pronouns can enter into this type of relations, e.g. *Mark gaf zijn eerste loon volledig uit, maar Evelyn stortte het op haar rekening. (English: Mark spent his entire first paycheck, but Evelyn deposited it into her account.)* Such a pronoun is known in literature as a paycheck pronoun (see for example (Cooper 1979), (Gardent 2000)).

- **part-whole/ element-set relations**, e.g. *Lehnert en Cardie komen. De twee zullen het hebben over anaforen resolutie. (English: Lehnert and Cardie come. Both will talk about anaphora resolution.)* Both "Lehnert" and "Cardie" are individuals and they are part of the group "de twee". Another example: *Hij kon zijn auto niet meer starten. De benzinetank was leeg. (English: He could not start the car. The gas tank was empty.)*
- (...)

When designing a manual for the encoding of anaphoric relations, it is necessary to first determine which anaphoric relations should be encoded. The annotation schemes of MUC-6 and MUC-7 (Hirschman and Chinchor 1998) and Davies et al. (1998), only cover the **identity relation**. They do not cover other types of coreference relations, such as set/subset, part/whole or type/token relations. Other schemes, such as the ones from Passoneau and Litman (1997) and Fligelstone (1990) encode more relations.

Furthermore, one can for example only encode the referential relations between **noun phrases**, or just between pronouns. In MUC-6 and MUC-7, the coreference relation is marked between noun phrases, including definite and indefinite noun phrases, nouns, different types of pronouns and proper names. It is however also possible to link relations involving verbs, wh-phrases, clauses, etc. (e.g. Passoneau and Litman (1997), Fligelstone (1990)).

A.1.3 Encoding coreferential relations

There are not many corpora available, which are annotated with anaphoric or coreferential links. For English, the data from the MUC coreference task are used for training and evaluating many systems. These MUC data sets, however, contain a rather limited number of words (12,400 words in MUC-6 and 19,000 words in MUC-7 for training) and there is still need for much more annotation efforts (as those described in Orasan (2000)). Other more recent important data sets for coreference resolution are the ACE (Automatic Content

Extraction) data sets[1], which provide more annotated data. For Dutch, there is as far as we know up to now no corpus available in which the anaphoric or coreferential relations are encoded. Coreferentially annotated corpora can be used to train and evaluate machine learning or statistical algorithms (e.g. Aone and Bennett (1995), Fisher et al. (1995), McCarthy and Lehnert (1995), McCarthy (1996), Ge et al. (1998), Cardie and Wagstaff (1999), Soon et al. (2001), Ng and Cardie (2002a,2002b,2002c) for coreference resolution. They can also be used for evaluation of knowledge-based coreference resolution systems and can also serve for linguistic research.

When designing a manual for the encoding of anaphoric relations, it is necessary to first determine which anaphoric relations should be encoded. One can for example only encode the referential relations between noun phrases (as in MUC-6 or MUC-7). It is however also possible to link relations involving verbs, wh-phrases, clauses, etc. In the annotation schemes for MUC-6 and MUC-7 (Hirschman and Chinchor 1998), only the "identity" relation for noun phrases is covered. It does not cover other types of coreference relations, such as set/subset, part/whole or type/token relations. For the development of the scheme for the annotation of Dutch newspaper text, we took the MUC-7 (Hirschman and Chinchor 1998) manual and the manual from Davies et al. (1998) as guidelines. In the annotation manual from Davies et al. (1998), the MUC coreference scheme is used as core scheme and an extended scheme also contains additional types of coreference. We also took into account the critical remarks from Kibble (2000) and van Deemter and Kibble (2000).

[1] See http://www.itl.nist.gov/iaui/894.01/tests/ace/index.htm for more information on these data sets

A.2 Annotation scheme

All types of NPs can enter into a coreference relation. (71) and (72) are some example sentences containing coreference relations.

(71) **Een van de sterkste stijgers binnen de DJ Stoxx50** is **ABN Amro**. De nettowinst van **de Nederlandse bank** daalde in het tweede kwartaal met 20%. Toch blijft **de bankgroep** sceptisch over het herstel van de economie.

English: One of the strongest gains in the DJ Stoxx50 is ABN Amro. The net profit of the Dutch bank dropped with 20% in the second quarter. Nevertheless, the bank group remains skeptical about the recovery of the economy.

Coreference annotation:
<**COREF ID = "1" MIN = "een"**>**Een van de sterkste stijgers binnen de DJ Stoxx50** is <**COREF ID = "2" TYPE = "IDENT" REF = "1"**> **ABN Amro** </**COREF**>. De nettowinst van <**COREF ID = "3" TYPE = "IDENT" REF = "2" MIN = "bank"**> **de Nederlandse bank** </**COREF**> daalde in het tweede kwartaal met 20%. Toch blijft <**COREF ID = "4" TYPE = "IDENT" REF = "3"**> **de bankgroep** </**COREF**> sceptisch over het herstel van de economie.

(72) In Duitsland maakte **autobouwer BMW** tweedekwartaalresultaten bekend. De nettowinst steeg en **het bedrijf** verwacht de doelstellingen voor 2002 te halen. **BMW** zakt 2,1% tot **EUR 38,68**.

English: In Germany, car manufacturer BMW announced second quarter results. The net profit increased and the company expects to achieve the objectives for 2002. BMW drops 2,1% to EUR 38,68.

Coreference annotation:
In Duitsland maakte <**COREF ID = "1" MIN = "BMW"**> **autobouwer BMW** </**COREF**> tweedekwartaalresultaten bekend. De nettowinst steeg en <**COREF ID = "2" TYPE = "IDENT" REF = "1"**> **het bedrijf** </**COREF**> verwacht de doelstellingen voor 2002 te halen. <**COREF ID = "3"**> **BMW** </**COREF**> zakt 2,1% tot <**COREF ID = "4" TYPE = "IDENT" REF = "3"**> **EUR 38,68** </**COREF**>.

> In this example, there are two coreference chains: one for "autobouwer BMW" and "het bedrijf" and a second chain with "BMW" and "EUR 38,68". The "BMW" with ID=1 and ID=3 are not identical. With the first "BMW", the company is meant. With the second "BMW", the stock option is meant, not the company.

The annotation of these two examples and all the following examples mainly follows the MUC guidelines. This means that all coreferences start with a <COREF> tag and are closed with a </COREF> close tag. The initial <COREF> tag contains additional information to the coreference: the ID of the coreference (ID), the type of coreference relation (TYPE), the ID of the entity referred to (REF) and the minimal tag of the coreference (MIN):

- **ID**: The "ID" is a unique ID given to the NP.
- **TYPE**: In the MUC annotation scheme, only one type of coreference relation is marked, viz. the identity relation ("IDENT"). We will also annotate this identity relation and other types of coreference relations will also be used in our annotation scheme. These other types will be explained later in this manual.
- **REF**: The "REF" attribute indicates that there is a coreference between two NPs. The "REF" attribute links the current NP referring to a previously mentioned NP. A sequence of NPs referring to each other is called a "coreference chain".
- **MIN**: The "MIN" string will in general be the head of the phrase. It indicates the minimum string that the system under evaluation must include in order to receive full credit for its output.

In the following examples, these attributes will be explained in more detail and a recapitulation of the different coreference attributes is given in SectionA.4.

A.2.1 Names and named entities

Names and named entities can all enter into coreference relations: names of companies (as in 71 and 72), organizations, persons, locations, dates, times, currency amounts, percentages, etc. Substrings of named entities are not marked: e.g. in (75), België is not marked separately. Dates are also marked as a whole.

(73) **De Unie van Zelfstandige Ondernemers (Unizo)** dient klacht tegen **Banksys**. Volgens **de Unizo** misbruikt **Banksys zijn** monopoliepositie.

English: The Unie van Zelfstandige Ondernemers (Unizo) brings an action against Banksys. According to Unizo, Banksys takes advantage of its monopoly position.

Coreference annotation:
<**COREF ID = "1"**> **De Unie van Zelfstandige Ondernemers** </**COREF**> (<**COREF ID = "2" TYPE = "IDENT" REF = "1"**> **Unizo** </**COREF**>) dient klacht tegen <**COREF ID = "3"**> **Banksys** </**COREF**>. Volgens <**COREF ID = "4" TYPE = "IDENT" REF = "2"**> **de Unizo** </**COREF**> misbruikt <**COREF ID = "5" TYPE = "IDENT" REF = "3"**> **Banksys** </**COREF**> <**COREF ID = "6" TYPE = "IDENT" REF = "5"**> **zijn** </**COREF**> monopoliepositie.

(74) **Marc Coenen** is benoemd tot **nethoofd van Studio Brussel**. **Coenen** stond 19 jaar geleden, samen met Hautekiet en Jan Schoukens, aan de wieg van **de jongerenmuziekzender**.

English: Marc Coenen has been appointed to head of Studio Brussel. 19 years ago, Coenen was one of the founders of the youth music station together with Hautekiet and Jan Schoukens.

Coreference annotation:
<**COREF ID = "1"**> **Marc Coenen** </**COREF**> is benoemd tot <**COREF ID = "2" TYPE = "IDENT" REF = "1" MIN = "nethoofd"**> **nethoofd van** <**COREF ID = "3"**> **Studio Brussel** </**COREF**> </**COREF**>. <**COREF ID = "4" REF = "IDENT" REF = "1"**> **Coenen** </**COREF**> stond 19 jaar geleden, samen met Hautekiet en Jan Schoukens, aan de wieg van <**COREF ID = "5" TYPE = "IDENT" REF = "3"**> **de jongerenmuziekzender** </**COREF**>.

(75) **Duitstalig België** worstelt met **zijn** identiteit.

English: The German speaking Community of Belgium wrestles with its identity.

Coreference annotation:

<**COREF ID = "1"**> **Duitstalig België** </**COREF**> worstelt met <**COREF ID = "2" TYPE = "IDENT" REF = "1"**> **zijn** </**COREF**> identiteit.

A.2.2 Pronouns

Personal/demonstrative/possessive/indefinite pronouns

Personal, demonstrative, possessive and indefinite pronouns can all enter into coreference relations. Some examples:

(76) **De regularisatieprocedure** startte begin 2000. **Zij** moest personen die al jaren illegaal in België verblijven de kans geven via een regularisatie wettelijk in België te wonen.

English: The regularization procedure was launched in the beginning of 2000. It was intended to enable persons which reside illegally in Belgium to legally live in Belgium through regularization.

Coreference annotation:
<**COREF ID = "1"**> **De regularisatieprocedure** </**COREF**> startte begin 2000. <**COREF ID = "2" TYPE = "IDENT" REF = "1"**> **Zij** </**COREF**>moest personen die al jaren illegaal in <**COREF ID = "3"**> **België** </**COREF**> België verblijven de kans geven via een regularisatie wettelijk in <**COREF ID = "4" TYPE = "IDENT" REF = "3"**> **België** </**COREF**> te wonen.

(77) **De Bank of Japan** heeft beslist **haar** rentepolitiek te behouden.

English: The Bank of Japan has decided to keep its interest rate policy.

Coreference annotation:
<**COREF ID = "1"**> **De Bank of Japan** </**COREF**> heeft beslist <**COREF ID = "2" TYPE = "IDENT" REF = "1"**> **haar** </**COREF**> rentepolitiek te behouden.

(78) Maar al snel bleek dat ook **circuits van mensenhandelaren** de procedure uitgekozen hadden om **hun** 'klanten' België binnen te loodsen.

English: But soon, it became clear that also circuits of human traffickers had picked the procedure to sneak their 'customers' into Belgium.

Coreference annotation:
Maar al snel bleek dat ook <**COREF ID = "1"**> **circuits van mensenhandelaren** </**COREF**> de procedure uitgekozen hadden om <**COREF ID = "2" TYPE = "IDENT" REF = "1"**> **hun** </**COREF**> 'klanten' België binnen te loodsen.

In some cases, pronouns may have no antecedent at all (e.g. 79) or they may also refer to something beyond our scope of annotation, such as a clausal construction (e.g. 80 and 81). In those cases, NO coreference is marked.

(79) **Het** regent pijpestelen.

English: It's raining cats and dogs.

(80) **Het herstel van de economische bedrijvigheid in eurozone zet zich voort, maar de onzekerheid van de kracht van het huidige herstel is groot. Dat** schrijft de Europese Centrale Bank (ECB) donderdag in het Maandbericht over augustus.

English: The recovery of the economic activity in the Eurozone persists, but the uncertainty about the strength of the current recovery remains large. This was written in the European Central Bank (ECB) monthly magazine of august.

(81) **Marc Coenen volgt Jan Hautekiet op als nethoofd van jongerenmuziekzender Studio Brussel**. **Dat** bevestigde Paul De Meulder.

English: Marc Coenen succeeds Jan Hautekiet as head of the youth music station Studio Brussel. This was confirmed by Paul De Meulder.

Reflexive pronouns

- Coreference annotation of reflexive pronouns if they denote an item in the world (e.g. 82).
- NO coreference annotation in the case of lexicalised reflexive pronouns (e.g. 83). Those pronouns do not refer to an argument and cannot be replaced by another NP.

(82) **Passagiers van de gekaapte vlucht 93 van United Airlines** offerden **zichzelf** op.

English: Passengers of the hijacked flight 93 of United Airlines sacrificed themselves.

COREFERENCE ANNOTATION:
<COREF ID = "1"> Passagiers van de gekaapte vlucht 93 van United Airlines </COREF> offerden **<COREF ID = "2" TYPE = "IDENT" REF = "1"> zichzelf </COREF>** op.

(83) De komende weken wijdt **Coenen**, net terug uit vakantie, **zich** volledig aan de Donna-evenementen.

English: During the next weeks, Coenen, who just returned from vacation, will commit himself to the Donna events.

Null pronouns

Null pronouns are NOT annotated for coreference. E.g.

(84) **Een passagier, de ondernemer Tom Burnett**, belde naar huis en ϕ vertelde dat er na stemming was besloten te pogen de drie kapers te overmeesteren.

English: A passenger, the entrepreneur Tom Burnett made a phone call home and ϕ told that after a vote it was decided to try to charge the three hijackers.

A.2.3 Conjoined noun phrases

When 2 or more NPs are conjoined or disjoined, it may be necessary to mark up the larger NP as well as the constituent NPs, depending on whether it is referred to later in the dialog.

(85) **Marc Coenen** volgt **Jan Hautekiet** op als nethoofd van jongerenmuziekzender Studio Brussel. **Coenen** stond 19 jaar geleden, samen met Schoukens en **Hautekiet**, aan de wieg van Studio Brussel.

English: Marc Coenen succeeds Jan Hautekiet as head of Studio Brussel. 19 years ago, Coenen was one of the founders of the youth music station together with Hautekiet and Jan Schoukens.

Coreference annotation:
<**COREF ID = "1"**> **Marc Coenen** </**COREF**> volgt <**COREF ID = "2"**> **Jan Hautekiet** </**COREF**> op als <**COREF ID = "3" TYPE = "IDENT" REF = "1"**> **nethoofd van** <**COREF ID = "4" MIN = "Studio Brussel"**> **jongerenmuziekzender Studio Brussel** </**COREF**> </**COREF**>. <**COREF ID = "5" TYPE = "IDENT" REF = "3"**> **Coenen** </**COREF**> stond 19 jaar geleden, samen met Schoukens en <**COREF ID = "6" TYPE = "IDENT" REF = "2"**> **Hautekiet** </**COREF**>, aan de wieg van <**COREF ID = "7" TYPE = "IDENT" REF = "4"**> **Studio Brussel** </**COREF**>.

(86) We hebben gisteren **Jan en Piet** ontmoet. **Piet** vertelde dat **ze** op weg waren naar een concert van Helmut Lotti. **Jan** had er duidelijk geen zin in.

English: Yesterday, we met Jan and Piet. Piet told us that they were on their way to a concert of Helmut Lotti. Jan apparently didn't feel like it.

Coreference annotation:
We hebben gisteren <**COREF ID = "1"**> <**COREF ID = "2"**> **Jan** </**COREF**> **en** <**COREF ID = "3"**> **Piet** </**COREF**> </**COREF**> gezien. <**COREF ID = "4" TYPE = "IDENT" REF = "3"**> Piet vertelde dat <**COREF ID = "5" TYPE = "IDENT" REF = "1"**> **ze** </**COREF**> op weg waren naar een concert van Helmut Lotti. <**COREF ID = "6" TYPE = "IDENT" REF = "2"**> **Jan** </**COREF**> had er duidelijk geen zin in.

A.2.4 NPs containing relative clauses

- Coreference annotation for NPs with restrictive relative clauses (as in 87).
- NO coreference annotation for NPs with non-restrictive relative clauses: (88).

(87) **Geruchten dat het toestel zou zijn neergehaald door de Amerikaanse luchtafweer**, zijn niet door ooggetuigen bevestigd. **Die geruch-**

ten werden dan ook snel afgedaan als nonsens.

English: Rumors that the plane was shot down by American anti-aircraft guns, are not confirmed by eye witnesses. These rumors were soon considered as nonsense.

COREFERENCE ANNOTATION:
<**COREF ID = "1" MIN = "geruchten"**> **Geruchten dat het toestel zou zijn neergehaald door de Amerikaanse luchtafweer** </**COREF**>, zijn niet door ooggetuigen bevestigd. <**COREF ID = "2" TYPE = "IDENT" REF = "1"**> **Die geruchten** </**COREF**> werden dan ook snel afgedaan als nonsens.

(88) **President Alejandro Toledo** reisde dit weekend naar Seattle voor een gesprek met **Microsoft topman Bill Gates**. **Gates**, die al jaren bevriend is met **Toledo**, investeerde onlangs zo'n 550.000 Dollar in Peru.

English: This weekend, president Alejandro Toledo traveled to Seattle to talk with Microsoft top executive Bill Gates. Gates, who has been close friends with Toledo for years, recently invested about 550.000 Dollar in Peru.

COREFERENCE ANNOTATION:
<**COREF ID = "1" MIN = "Alejandro Toledo"**> **President Alejandro Toledo** </**COREF**> reisde dit weekend naar Seattle voor een gesprek met <**COREF ID = "2" MIN = "Bill Gates"**> **Microsoft topman Bill Gates** </**COREF**>. <**COREF ID = "3" TYPE = "IDENT" REF = "2"**> **Gates** </**COREF**>, die al jaren bevriend is met <**COREF ID = "4" TYPE = "IDENT" REF = "1"**> **Toledo** </**COREF**>, investeerde onlangs zo'n 550.000 Dollar in Peru.

A.2.5 Other phrases without a head noun

= phrases with nominalized adjectives, infinitives, gerunds or quantifiers as heads can also enter into coreference relations.

(89) **Het eten van 2 stukken fruit per dag** wordt nog te weinig gestimuleerd. **Het** is nochtans heel goed voor de gezondheid.

English: (In English, the Dutch nominalized infinitive is translated as a gerund.) Eating two pieces of fruit each day is still under-stimulated. It is however very healthy.

Coreference annotation:
<**COREF ID = "1" MIN = "eten"**> **Het eten van 2 stukken fruit per dag** </**COREF**> wordt nog te weinig gestimuleerd. <**COREF ID = "2" TYPE = "IDENT" REF = "1"**> **Het** </**COREF**> is nochtans heel goed voor de gezondheid.

A.3 Special cases

A.3.1 Bound anaphors

We will also mark a coreference relation between a bound anaphor and the NP which binds it (see 90).

(90) **Niemand** verliest graag **zijn** job.

English: Nobody likes to lose his job.

Coreference annotation:
<**COREF ID = "1"**> **Niemand** </**COREF**> verliest graag <**COREF ID = "2" IDENT = "BOUND" REF = "1"**> **zijn** </**COREF**> job.

(91) **Geen enkel kind** zal toegeven dat **het** moe is.

English: No child will admit that it is tired.

Coreference annotation:
<**COREF ID = "1"**> **Geen enkel kind** </**COREF**> zal toegeven dat <**COREF ID = "2" IDENT = "BOUND" REF = "1"**> **het** </**COREF**> moe is.

For the annotations of these bound anaphors, we define a new type of relation (as also proposed by Davies et al. (1998)): "BOUND".

Some more examples:

(92) **Most linguists** prefer **their** own parsers.

(93) **Every TV network** reported **its** profits.

A.3.2 "Paycheck" pronouns

(94) The man who gave **his paycheck** to his wife was wiser than the man who gave **it** to his mistress.

COREFERENCE ANNOTATION:
The man who gave <**COREF ID = "1"**> **his paycheck** <**/COREF**> to his wife was wiser than the man who gave <**COREF ID = "2" TYPE = "ISA" REF = "1"**> **it** <**/COREF**> to his mistress.

The paycheck pronouns owe their name to the classical example in (94). A similar relation is expressed in (95):

(95) Ik verkies **de rode auto** boven **de grijze**.

English: I prefer the red car to the gray one.

COREFERENCE ANNOTATION:
Ik verkies <**COREF ID = "1" MIN = "auto"**> **de rode auto** <**/COREF**> boven <**COREF ID = "2" TYPE = "ISA" REF = "1"**> **de grijze** <**/COREF**>.

In both examples, "it" in (94) and "de grijze" in (95) do not refer to the same extralinguistic entity as their respective antecedents "his paycheck" and "de rode auto". So there is no identity relation between both NPs. In order to capture this type of relationships, we will follow the definition of Hirst (1981) and distinguish between **identity of sense anaphora (ISA)** and **identity of reference anaphora (IRA)**. An IRA (in the MUC and in our annotation scheme: "IDENT") is an anaphor which denotes the same entity as its antecedent. An ISA is an anaphor (as in 94 and 95) which denotes not the same entity as its antecedent, but one of a similar description.

A.3.3 Appositions

For the annotation of appositions, we loosely followed the instructions from Hirschman and Chinchor (1998) and Davies et al. (1998). Annotation instruc-

tions:

- The MUC manual proposes to tag the NP as a whole as well as any separate NP contained in the appositive clauses, if the appositive clause is contiguous to the NP. The appositions refer to the complete NP. Also indefinite appositions are marked. We will not follow this proposal and tag both NPs of the apposition as separate NPs.

  (96) **De Franse bouwgroep Vinci,** ***moeder van het Belgische CFE***, boekte over de eerste helft van 2002 een 2,5% hogere omzet.

  English: The French construction group Vinci, mother company of the Belgian CFE, had a 2.5% turnover increase in the first half of 2002.

  Coreference annotation:
  <COREF ID = "1" MIN = "Vinci"> De Franse bouwgroep Vinci </COREF>, **<COREF ID = "2" TYPE = "IDENT" REF = "1" MIN = "moeder"> moeder van het Belgische CFE</COREF>**, boekte over de eerste helft van 2002 een 2,5% hogere omzet.

  (97) **Michel Counson,** ***de voorzitter van de raad van bestuur***, verminderde **zijn** belang met 0,2%.

  English: Michel Counson, the president of the Board of Directors, reduced its interests by 0.2%.

  Coreference annotation:
  <COREF ID = "1"> Michel Counson </COREF>, **<COREF ID = "2" TYPE = "IDENT" REF = "1" MIN = "voorzitter"> de voorzitter van de raad van bestuur</COREF>**, verminderde **<COREF ID = "3" TYPE = "IDENT" REF = "2"> zijn </COREF>** belang met 0,2%.

  (98) **Marc Leemans,** ***analist bij KBC Securities***, vermoedt dat de impact op de resultaten van Umicore beperkt blijft.

  English: Marc Leemans, KBC Securities analist, expects that the impact on the results of Umicore remains limited.

COREFERENCE ANNOTATION:
<COREF ID = "1"> Marc Leemans </COREF>, **<COREF ID = "2" TYPE = "IDENT" REF = "1" MIN = "analist"> analist bij KBC Securities </COREF>**, vermoedt dat de impact op de resultaten van Umicore beperkt blijft.

(99) Unizo dient klacht in tegen **Banksys, *de beheerder van het elektronisch betaalverkeer***.

English: Unizo sues Banksys, the administrator of electronic payments.

COREFERENCE ANNOTATION:
Unizo dient klacht in tegen **<COREF ID = "1"> Banksys </COREF>**, **<COREF ID = "2" TYPE = "IDENT" REF = "1" MIN = "beheerder"> de beheerder van het elektronisch betaalverkeer </COREF>**.

- In case of restrictive appositions, mark only the NP as a whole.

(100) **De Nederlanse bankgroep ABN AMRO** heeft over het tweede kwartaal van 2002 een nettowinst behaald van 534 miljoen euro.
English: The Dutch banking group ABN AMRO (...)

(101) **De beeldvormingsgroep Barco** koopt een deel van de activa van het Duitse Tan over van de familie Tan.
English: The visualization group Barco (...)

(102) De kwartaalresultaten van **het chemiebedrijf BASF** voldoen aan de verwachtingen van de analisten.
English: The quarter results of the chemical company BASF (...)

- According to the MUC guidelines, appositional phrases are NOT marked when they are negative or when there is only partial overlap of sets. We decided that also negative information is information and we therefore also mark these appositional phrases.

(103) **Karel Degucht, *niet meteen een toonbeeld van bescheidenheid*** ...
English: Karel Degucht, not exactly a model of modesty ....

(104) **De criminelen, *vaak genaturaliseerde Belgen*** ...
English: The criminals, often naturalized Belgians ...

A.3.4 Predicate nominals

In MUC (Hirschman and Chinchor 1998), all predicate nominals can enter into coreference relations. Davies et al. (1998), however, claim that predicative noun phrases (often indefinite NPs) cannot be considered to refer. This approach of integrating predicate nominals into coreference relations has also been criticized by van Deemter and Kibble (2000). For our annotations of Dutch sentences, we will follow the MUC annotations and also allow definite predicate nominals in coreference relations (see 105, 106 and 107).

(105) **Het mediabedrijf Vivendi Universal** is **de tweede sterktste stijger binnen de DJ Stoxx50**.

English: The media company Vivendi Universal is the second largest performer in the DJ Stoxx50.

Coreference annotation:
<COREF ID = "1" MIN = "Vivendi Universal"> Het mediabedrijf Vivendi Universal </COREF> is **<COREF ID = "2" TYPE = "IDENT" REF = "1" MIN = "stijger"> de tweede sterktste stijger binnen de DJ Stoxx50 </COREF>**.

(106) **Vivendi Universal** is **de tweede grootste producent van videospelletjes voor PC in de wereld**.

English: Vivendi Universal is the world's second largest video game maker for PC.

Coreference annotation:
<COREF ID = "1"> Vivendi Universal </COREF> is **<COREF ID = "2" TYPE = "IDENT" REF = "1" MIN = "producent"> de tweede grootste producent van videospelletjes voor PC in de wereld </COREF>**.

(107) **Kim Gevaert** is **de eerste Belgische vrouw die een medaille verovert op een Europees kampioenschap**.

English: Kim Gevaert is the first Belgian woman which wins a medal at a European championship.

Coreference annotation:

<COREF ID = "1"> Kim Gevaert </COREF> is <COREF ID = "2" TYPE = "IDENT" REF = "1" MIN = "vrouw"> de eerste Belgische vrouw die een medaille verovert op een Europees kampioenschap </COREF>.

We will NOT record coreference in case of only the possibility of coreference between two NPs (as in 108).

(108) **Filip Dewinter** had wel eens **de nieuwe burgemeester van Antwerpen** kunnen worden.

English: Filip Dewinter could have become the new mayor of Antwerp.

A.3.5 Time-dependent identity

In the MUC annotation scheme (Hirschman and Chinchor 1998), two NPs are recorded as coreferential if the text asserts them to be coreferential at ANY TIME. This implies that in (109), there will be a coreference chain between "Guy Vanhengel", "Vlaams minister voor Sport en Brusselse Aangelegenheden" and "Brussels minister van Financiën, Begroting en Openbaar Ambt". In (110), coreference is marked between "Bert Degraeve", "gedelegeerd bestuurder van de VRT" and "chief financial and administration manager".

(109) Niet alleen wordt **Guy Vanhengel Vlaams minister voor Sport en Brusselse Aangelegenheden**, tevens blijft hij ook nog eens **Brussels minister van Financiën, Begroting en Openbaar Ambt**.

English: Guy Vanhengel does not only become Flemish minister of Sports and Brussels Affairs, he also remains the Brussels Minister of Finance, Budget and Public Affairs.

Coreference annotation:
Niet alleen wordt **<COREF ID = "1"> Guy Vanhengel </COREF>** **<COREF ID = "2" TYPE = "IDENT" REF = "1" MIN = "minister"> Vlaams minister voor Sport en Brusselse Aangelegenheden </COREF>**, tevens blijft hij ook nog eens **<COREF ID = "3" TYPE = "IDENT" REF = "1" MIN = "minister"> Brussels minister van Financiën, Begroting en Openbaar Ambt </COREF>**.

(110) **Bert Degraeve**, die tot voor kort **gedelegeerd bestuurder van de VRT** was, gaat aan de slag bij staaldraad- en coatingsproducent Bekaert als **chief financial and administration manager**.

English: Bert Degraeve, until recently managing director of the VRT, starts to work at steel wire and coating producer Bekaert as chief financial and administration manager.

Coreference annotation:
<**COREF ID = "1"**> **Bert Degraeve** </**COREF**>, die tot voor kort <**COREF ID = "2" TYPE = "IDENT" REF = "1" MIN = "bestuurder" TIME = "1"**> **gedelegeerd bestuurder van de VRT** </**COREF**> was, gaat aan de slag bij staaldraad- en coatingsproducent Bekaert als <**COREF ID = "3" TYPE = "IDENT" REF = "1" MIN = "manager" TIME = "2"**> **chief financial and administration manager** </**COREF**>.

Problematic in this MUC-approach (see also van Deemter and Kibble (2000)) is that coreference is agreed to be an equivalence relation. This implies that in (109), "gedelegeerd bestuurder van de VRT" and "chief financial and administration manager" are the same person, namely Bert Degraeve. This is clearly not the case. In (110), Guy Vanhengel performs both functions at the same time and so there is indeed an equivalence between all parts in the coreference chain. For the annotation of these time-dependent identities, we will follow the MUC-approach, but we will also take into account the criticism from van Deemter and Kibble (2000) and add a time-indication in the annotation of these NPs (expressed in the "TIME" attribute).

A similar example is given in (111), but a different annotation approach is proposed in the MUC scheme:

(111) Per aandeel steeg **de nettowinst van de halfgeleiderproducent Melexis** van **EUR 0,11** tot **0,12**.

English: The net profit of semiconductor producer Melexis increased from EUR 0.11 to EUR 0.12 per share.

Coreference annotation:
Per aandeel steeg <**COREF ID = "1" MIN = "nettowinst"**> **de nettowinst van de halfgeleiderproducent Melexis** </**COREF**> van <**COREF ID = "2" TYPE = "IDENT" REF = "1" TIME = "1"**> **EUR 0,11** </**COREF**> tot <**COREF ID = "3" TYPE = "IDENT" REF = "1" TIME = "2"**> **0,12** </**COREF**>.

In the MUC annotation scheme, only **the most recent value** (in the case of (111): "0,12") is marked as coreferential with "de nettowinst van de halfgeleiderproducent Melexis". And "EUR 0,11" is put into a separate coreference class. Unlike in the MUC scheme, we believe that the sentences in (109), (110) and (111) are similar and therefore we will also annotate them in a similar way (as also proposed by Davies et al. (1998)).

A.3.6 Metonymy

= a rhetorical substitution of one thing for another based on their association or proximity, e.g. "monarch" and "crown" are metonyms. Following the MUC annotation scheme, we will also annotate coreference relations between metonyms.

(112) **Boudewijn** moest in die dagen niet lang zoeken naar kanalen om zijn macht in daden om te zetten. Het lijdt geen twijfel dat **Laken** gedurende de hele periode 1960-1961 **zijn** rol in de coulissen heeft gespeeld. **Het paleis** is nooit veel meer, maar zeker nooit minder geweest dan **de exponent van de Belgische heersende klasse in haar conservatisme, in haar katholicisme, en met haar financieel- economische macht.**

English: In those days, Boudewijn did not have to look long for ways to put his power into action. It is beyond doubt that Laken played its role behind the scenes during the whole period 1960-1961. The royal palace has never been much more, but has certainly never been less than the exponent of the Belgian ruling class in all its conservatism, in its catholicism and with is financial-economical power.

COREFERENCE ANNOTATION:

<COREF ID = "1"> Boudewijn </COREF> moest in die dagen niet lang zoeken naar kanalen om zijn macht in daden om te zetten. Het lijdt geen twijfel dat **<COREF ID = "2" TYPE = "IDENT" REF = "1"> Laken </COREF>** gedurende de hele periode 1960-1961 **<COREF ID = "3" TYPE = "IDENT" REF = "2"> zijn </COREF>** rol in de coulissen heeft gespeeld. **<COREF ID = "4" TYPE = "IDENT" REF = "3"> Het paleis </COREF>** is nooit veel meer, maar zeker nooit minder geweest dan **<COREF ID = "5" TYPE = "IDENT" REF = "4"> de exponent van <COREF ID = "6"> de Belgische heersende klasse </COREF> in <COREF ID = "7" TYPE = "IDENT" REF = "6"> haar </COREF> conservatisme, in <COREF ID = "8" TYPE = "IDENT" REF**

= "6"> haar </COREF> katholicisme, en met <COREF ID = "9" TYPE = "IDENT" REF = "6"> haar </COREF> financieel-economische macht </COREF>.

A.3.7 Set relations, possessive relations, discontinuous NPs, etc.

For the time being, no coreference annotation is given for all these types of relations.

A.4 The coreference attributes

In Section A.2, we already gave a brief introduction to the different types of attributes in the coreference annotation. We showed in numerous examples that all coreferences start with a <COREF> tag and are closed with a </COREF> close tag. The initial <COREF> tag contains additional information to the coreference: the ID of the coreference ("ID"), the type of coreference relation ("TYPE"), the ID of the entity referred to ("REF"), the minimal tag of the coreference ("MIN") and (if necessary) time information ("TIME"):

- **ID**: The "ID" is a unique ID given to the NP.
- **TYPE**: In the MUC annotation scheme, only one type of coreference relation is marked, viz. the identity relation ("IDENT"). We will also annotate this identity relation and other types of coreference relations will also be used in our annotation scheme. These other types include: BOUND (see examples 90 and 91) and ISA (see 94 and 95).
- **REF**: The "REF" attribute indicates that there is a coreference between two NPs. The "REF" attribute links the current NP referring to a previously mentioned NP. A sequence of NPs referring to each other is called a "coreference chain". There are different possibilities to mark the coreference links. If there is, for example, a coreference relation between "A", "B" and "C", this relationship can be annotated in two ways:
  - both B and C refer to A
  - or C refers to B and B refers to A.
- **MIN**: The "MIN" string will in general be the head of the phrase. It indicates the minimum string that the system under evaluation must include

in order to receive full credit for its output. The "MIN" attribute includes:

  - the main noun without its left and right modifiers, e.g. (95), (97), (98), and (99).
  - In the case of names, the entire name is marked without any personal titles or modifiers, e.g. (96).
  - Dates, currency amounts and percentages are taken as a whole, e.g. (111).
  - In headless constructions, the head is the last token of the NP preceding any prepositional phrases, relative clauses and other 'right' modifiers, e.g. (89).
  - If the maximal NP consists of a single head or a head preceded by an article, the "MIN" need not be marked.
  - In the case of conjunctions, the minimal string begins at the minimal phrase for the first conjunct and includes everything up to the end of the minimal phrase for the last conjunct.

- **TIME**: The "TIME" attribute is used in the case of time-dependent identity (see A.3.5).

A.5 The ALEMBIC Workbench

The Alembic Workbench is the annotation environment we will use for the annotation of the coreferences in Dutch texts. More information on this workbench can be found at http://www.mitre.org/tech/alembic-workbench.

A.5.1 Starting the Alembic Workbench

- Go to the alembic directory
- Typ **csh**.
- Typ **source awb.cshrc**.
- Start the workbench by typing **awb**.

A.5.2 5 Menus

When the Alembic Workbench GUI is invoked, it appears as a text display with 5 menus: (1) File, (2) Tag (:current annotation type), (3) Options, (4) Utilities and (5) Help. We will now discuss the different menus in more detail. We will only discuss the options which will be used for our annotation needs.

The File Menu

- **Open Document (Latin-1)**: When you click this option, a new display is opened, showing the texts in the directory. In this directory, you choose a file by double clicking on it. The press "open". If a file is being opened in the workbench for the first time, a preprocessing dialog is launched from which preprocessing and normalization options are chosen. Choose "No normalization" and then press "Do It". This causes a new window to be opened with the title "Tagset Contents". Press "Load File" in this window. The selected file is then opened in the Workbench and a copy of the original file is saved in a backup file with extension ".number" in the /home/hoste/software/alembic/data directory.

  If the file has previously been opened in the workbench, the preprocessing dialog will not be launched. Instead, files created when the document was first opened in the Workbench are consulted. These files are in an internal format called the Parallel Tag File Format (PTF).

- **Close**: Closes a document after it has been opened in the Alembic Workbench. If the file has been changed, a Save dialog will be launched.

- **View Source SGML of Current Document**: The user can choose to view the current document or another document. This option launches an SGML-encoded document viewer that allows the user to choose colors with which to display annotations.

- **Save**: Save changes made to file.

- **Save as**: Save changes made to files under a different file name.

- **Recover Original File**: Access the original unmodified version of the file currently loaded in the interface and enables the user to revert to it, if necessary.

- **Quit**: With this option, the user can quit the workbench.

The Tag Menu

The Tag Menu is colorized and contains abbreviations that correspond to SGML annotations. The tag menu lists those tags that can be used for the annotation of the texts.

The Options Menu

In the Options Menu, only the option "Coreference" is important for our annotation needs. This option makes available the options:

- **Show Coreference Targets**: launches a scrolling list box that contains all strings denoted by the user as coreference referents.
- **Highlight Coreference Chain**: allows the user to selectively highlight one chain of coreference.
- **Remove Highlighting of Coreference Chain**: removes the highlighting from the chain.
- **Hide Coreference Chains**: hides chains selected by the user which can be restored with the option "Restore hidden Coreference Chains".

The Utilities Menu

The only option of use for the annotators is "Find in current document": this utility makes available a search mechanism that can be used to perform searches either on text strings or on tags. The search utility also includes a Tag Selection option from which the user can choose to add tags to those text strings found in the document.

The Help Menu

The Help Menu provides information on different topics.

A.6 How to annotate?

A.6.1 Annotation procedure

The Dutch newspaper texts will all be annotated by two annotators from a pool of five native speakers with a linguistic background. After the individual coreference annotation by both annotators, they will verify all annotations together in order to reach a consensus annotation.

A.6.2 Selecting phrases

For the selection of NPs, one can use the left and the right mouse button.

- **Left mouse button**: selects the word under the cursor and each mouse click causes an additional word to the left to be selected.
- **Right mouse button**: selects the word under the cursor and each mouse click causes a additional word to the right to be selected.

When selecting a certain NP, that text is annotated with the tag which is selected in the Tag Menu. If, for example, you want to tag an NP and that NP is mentioned for the first time, the following procedure has to be followed:

- Select in the Tag Menu the tag "Tag:Initial Mention".
- Select the NP that has to be tagged with this tag (left or right mouse button).
- **Confirm** the tagging of this selection by pressing the **middle mouse button**.

If you want to **change** the tag of a certain selection:
press control-shift left mouse button and then select a new tag for this selection

If you want to **delete** this selection:
press control-shift left mouse button and then press "delete" or "backspace".

A.6.3 Different coreference tags

Coreference markup must be transitive and symmetric. If A, B and C are participating in a coreference chain, then it holds that A=B, B=C and A=C.

As a result, coreference chains can be linked in any order according to the user's preference. At the bottom from the workbench window, the SGML tag is displayed. The tool automatically assigns values to the ID (an uniquely assigned number) and REF (the ID number of the base expression to which the current NP refers) attributes.

- **Initial Mention** is selected if the NP is mentioned for the first time.
- **Initial Mention w/ MIN from Selection** is selected if the NP is mentioned for the first time and if the NP contains a MIN attribute. After selecting this tag, the text *select area for MIN attribute* appears in the lower right corner of the workbench window and the selection in the text is blinking. With the left or right mouse button, you can select the head in this blinking NP and you can confirm this selection by pressing on the middle mouse button.
- **Coref of IDENT/BOUND/ISA type** is selected if the NP refers to a previously marked NP in the text. After selecting this tag, the text *Click <button-1> on reference COREF tag (for REF attribute)* appears in the lower right corner of the workbench window and the selection in the text is blinking. So the reference is established by clicking with the first mouse button on the NP referred to by the blinking NP.
- **Coref of IDENT/BOUND/ISA type w/ MIN from Selection** is selected if the NP refers to a previously marked NP in the text and if the current NP has a MIN attribute.
- **Coref of IDENT/BOUND/ISA type w/ TIME attribute**
- **Coref of IDENT/BOUND/ISA type w/ MIN from Selection w/ TIME attribute**

A.7 An example annotation

(Financieel Economische Tijd, 14/08/2002)

CBF waarschuwt voor activiteiten Cambridge International

De Commissie voor het Bank- en Financiewezen (CBF) waarschuwt voor de activiteiten van de Italiaanse vennootschap Cambridge International S.r.I. Die zou geen vergunning hebben om beleggingsdiensten aan te bieden in België.

Cambridge International, met maatschappelijke zetel in Milaan, zou beleggingsdiensten met betrekking tot financile instrumenten aanbieden aan het Belgische publiek. Maar de CBF waarschuwt ervoor dat de vennootschap niet over de nodige vergunning beschikt om zo'n diensten in België aan te bieden.

English: CBF warns for activities Cambridge International. The Commissie voor het Bank- en Financiewezen (CBF) warns for the activities of the Italian company Cambridge International S.r.I. It does not have a license to offer investment services in Belgium. Cambridge International, which has its seat in Milan, offers investment services with respect to financial instruments to the Belgian people. But the CBF warns that the company does not have the necessary license to offer this type of services in Belgium.

Coreference annotation:

<COREF ID = "1"> CBF </COREF> waarschuwt voor **<COREF ID = "2"> activiteiten </COREF> <COREF ID = "3"> Cambridge International </COREF>**

<COREF ID = "4" TYPE = "IDENT" REF = "1"> De Commissie voor het Bank- en Financiewezen <COREF> (**<COREF ID = "5" TYPE = "IDENT" REF = "4"> CBF </COREF>**) waarschuwt voor **<COREF ID = "6" TYPE = "IDENT" REF = "2"> de activiteiten <COREF>** van **<COREF ID = "7" TYPE = "IDENT" REF = "3" MIN = "Cambridge International S.r.I.> de Italiaanse vennootschap Cambridge International S.r.I. </COREF>**

<COREF ID = "8" TYPE = "IDENT" REF = "7"> Die </COREF> zou geen vergunning hebben om **<COREF ID = "9" TYPE = "IDENT" REF = "6"> beleggingsdiensten </COREF>** aan te bieden in **<COREF ID = "10"> België </COREF>**.

<COREF ID = "11" TYPE = "IDENT" REF = "8"> Cambridge International </COREF>, met maatschappelijke zetel in Milaan, zou **<COREF ID = "12" TYPE = "IDENT" REF = "9"> beleggingsdiensten met betrekking tot financiële instrumenten </COREF>** aanbieden aan **<COREF ID = "13" TYPE = "IDENT" REF = "10"> het Belgische publiek </COREF>**

Maar <**COREF ID = "14" TYPE = "IDENT" REF = "5"**> **de CBF** </**COREF**> waarschuwt ervoor dat <**COREF ID = "15" TYPE = "IDENT" REF = "11"**> **de vennootschap** </**COREF**> niet over de nodige vergunning beschikt om <**COREF ID = "16" TYPE = "IDENT" REF = "13"**> **zo'n diensten** </**COREF**> in <**COREF ID = "17" TYPE = "IDENT" REF = "13"**> **België** </**COREF**> aan te bieden.

APPENDIX B

Ripper rules for the MUC-6 "Proper nouns" data set

This Appendix contains the RIPPER rules learned after extensive feature selection and parameter optimization by a genetic algorithm. The following lines represent the Ripper rules that are learned on the basis of the "Proper nouns" MUC-6 training material. The numbers between brackets at the end of the line represent the number of instances which are correctly covered by the rule, followed by the number of instances which are falsely classified through application of the rule.

An instance is classified as positive ...

- if there is a complete match between both NPs and if the semantic class of the candidate anaphor is not undefined (1243/44)

- if there is a partial match and a complete match between both NPs, if the candidate antecedent is not an object (705/47)

- if there is a partial match and a complete match between both NPs, if third word to the left of the candidate anaphor is not "how", if the first word to the left of the candidate anaphor is not "for", if the POS of the

first and third word to the right of the candidate anaphor is not "IN", if the candidate anaphor is not a subject and if the candidate antecedent is not a subject nor an object (431/2)

- if there is a partial match and a complete match between both NPs, if the second POS to the left of the candidate antecedent is not a common noun, if the second word to the left of the candidate anaphor is not "understood" nor "n't" (304/25)

- if there is a partial match and a complete match between both NPs, if the first POS to the right of the candidate anaphor is not "VBZ", if the second word to the left of the candidate anaphor is not "Icahn" nor "shares", if the first POS to the left is not a common noun, if the third word to the left is not "injunction" (365/0)

- if there is a partial match and a complete match between both NPs, if the third POS to the right is a common noun and if the candidate anaphor is a subject (10/0)

- if there is a partial match and number agreement between both NPs, if the candidate antecedent is not a subject, if the candidate antecedent is not an object (149/2)

- if there is a partial match and number agreement between both NPs, if the candidate antecedent and the candidate anaphor both have an undefined semantic class, if the first word to the left of the candidate anaphor is not "of", if the third word to the left of the anaphor is not "group", if the second POS to the left of the candidate anaphor is a common noun, if the first POS to the left of the candidate anaphor is a preposition and if the candidate antecedent is not a subject nor an object (14/1)

- if there is a partial match and number agreement between both NPs, if the candidate anaphor is an apposition, if the candidate anaphor has an undefined semantic class and if the candidate antecedent is not a subject nor an object (62/5)

- if there is a partial match and number agreement between both NPs, if there is no complete match, if the candidate anaphor is a "I-ORG" named

entity, if the candidate anaphor has an undefined semantic class, if the first word to the left of the candidate anaphor is not "with", if the third POS to the right of the candidate anaphor is not a full stop, if the second word to the left of the anaphor is not a comma and if the candidate antecedent is not a subject nor an object (72/18)

- if there is a partial match and number agreement between both NPs, if the candidate anaphor is no apposition, if the candidate anaphor is a person, if the candidate antecedent is neutral, if the third word to the left of the candidate anaphor is not a comma, if the third POS to the left of the candidate anaphor is not a proper noun, if the candidate anaphor is a subject and if the candidate antecedent is not a subject nor an object (89/6)

- if there is a partial match and number agreement between both NPs, if the candidate anaphor is an apposition and if the candidate anaphor is male (96/0)

- if there is a partial match, a complete match and number agreement between both NPs, if the second POS to the right of the candidate anaphor is a proper noun and if the first word to the right of the candidate anaphor is not "and" (15/0)

- if there is a partial match and number agreement between both NPs, if the candidate anaphor is a person, if the candidate antecedent is not male, if the candidate anaphor is no apposition, if the third POS to the left of the candidate anaphor is not a proper noun, if the first word to the right of the right of the candidate anaphor is a right quote, if the sentence distance between both NPs is more than three sentences (78/23)

- if there is a partial match, a complete match and number agreement between both NPs, if the semantic class of the candidate anaphor is undefined, is the semantic class of the antecedent is not male, if the semantic class of the candidate anaphor is undefined, if the semantic class of the candidate antecedent is not male, if the distance between both NPs is less than three sentences and if the candidate antecedent is not a subject nor an object (19/0)

- if there is a partial match, if the candidate anaphor and candidate antecedent both have a defined semantic class, if the candidate antecedent

is an organization, if the candidate antecedent is no definite NP (89/17)

- if there is a partial match and number agreement between both NPs, if the candidate anaphor is a person, if the second word to the left of the candidate anaphor is not a proper noun, if the candidate antecedent is male, if the candidate anaphor is no apposition, if the second NP to the left of the candidate anaphor is a plural noun, if the first NP to the left of the candidate anaphor is no preposition and if the sentence distance between both NPs is less than three sentences (15/1)

- if there is a partial match and number agreement between both NPs, if both NPs are synonyms, if both NPs are a person, if both NPs are a subject and if the second POS to the right of the candidate anaphor is not a determiner (39/1)

- if there is a partial match and number agreement between both NPs, if the candidate anaphor is a "I-PER" named entity and if the candidate anaphor is male (63/40)

- if there is a partial match and number agreement between both NPs, if the candidate anaphor is an apposition, if the semantic class of the candidate anaphor is undefined, if the candidate antecedent is neutral and if the candidate antecedent is no subject (35/2)

- if there is a partial match, a complete match and number agreement between both NPs, if the first POS to the left of the candidate anaphor is a preposition and if the first word to the left of the candidate anaphor is "by" (14/0)

- if there is a partial match, if the candidate antecedent is neutral, if the semantic class of the candidate anaphor is undefined, if the candidate antecedent is not a subject and if the POS of the second word to the right of the candidate anaphor is an adverb (24/17)

All other NPs are classified as negative (68383/1970).

APPENDIX C

Three MUC-7 documents for which a qualitative error analysis has been carried out

This Appendix contains three texts from the MUC-7 test data. The qualitative error analysis in Chapter 8 is based on these three documents.

1. <DOC>
   <DOCID> nyt960108.0668 </DOCID>
   <COREF ID = "1"> Loral Space </COREF>
   <COREF ID = "79"> 01-08 </COREF>
   <COREF ID = "0" TYPE = "IDENT" REF = "1"> Loral Space </COREF>
   <COREF ID = "17" MIN = "deal"> <COREF ID = "3"> Loral </COREF>
   deal </COREF> to aid <COREF ID = "12"> Globalstar </COREF>
   (lb)
   By LAURENCE ZUCKERMAN
   c . 1996 N.Y. Times News Service
   One reason <COREF ID = "8"> Lockheed Martin Corp. </COREF>
   did not announce a full acquisition of <COREF ID = "2" TYPE =
   "IDENT" REF = "3"> Loral Corp. </COREF> on <COREF ID =
   "19"> Monday </COREF>, according to <COREF ID = "6" MIN =
   "Bernard Schwartz"> Bernard Schwartz, <COREF ID = "5" TYPE =

"IDENT" REF = "6" MIN = "chairman"> <COREF ID = "4" TYPE = "IDENT" REF = "2"> Loral </COREF> 's chairman </COREF>, </COREF> was that <COREF ID = "7" TYPE = "IDENT" REF = "8"> Lockheed </COREF> could not meet the price <COREF ID = "9" TYPE = "IDENT" REF = "5"> he </COREF> had placed on <COREF ID = "25" MIN = "ownership"> <COREF ID = "10" TYPE = "IDENT" REF = "4"> Loral </COREF> 's 31 percent ownership of <COREF ID = "11" TYPE = "IDENT" REF = "12"> Globalstar Telecommunications Ltd. </COREF> </COREF> <COREF ID = "13" TYPE = "IDENT" REF = "11"> Globalstar </COREF> plans to provide <COREF ID = "51" MIN = "service"> telephone service </COREF> by bouncing signals off 48 low-orbiting satellites. But with no customers expected until 1998, the need for nearly $ 2 billion in investment and numerous competitors lurking in the shadows, <COREF ID = "14" TYPE = "IDENT" REF = "13"> Globalstar </COREF> 's prospects would not appear to be valuable to the average Lockheed shareholder. Still, <COREF ID = "15" TYPE = "IDENT" REF = "9"> Schwartz </COREF> feels differently, and so now do many investors. News of <COREF ID = "16" TYPE = "IDENT" REF = "17" MIN = "deal"> <COREF ID = "18" TYPE = "IDENT" REF = "19"> Monday </COREF> 's deal </COREF>, in which <COREF ID = "20" TYPE = "IDENT" REF = "7"> Lockheed </COREF> will buy most of <COREF ID = "21" TYPE = "IDENT" REF = "10"> Loral </COREF> 's military businesses and invest $ 344 million in <COREF ID = "22" TYPE = "IDENT" REF = "0" MIN = "Loral Space and Communications Corp."> Loral Space and Communications Corp., <COREF ID = "23" TYPE = "IDENT" REF = "22" MIN = "company"> a new company whose <COREF ID = "24" TYPE = "IDENT" REF = "25" MIN = "holding"> principal holding </COREF> will be <COREF ID = "27" TYPE = "IDENT" REF = "24" MIN = "interest"> <COREF ID = "26" TYPE = "IDENT" REF = "21"> Loral </COREF> 's interest in <COREF ID = "28" TYPE = "IDENT" REF = "14"> Globalstar </COREF> </COREF> </COREF>, </COREF> sent <COREF ID = "29" TYPE = "IDENT" REF = "28"> Globalstar </COREF> 's own shares soaring $6.375, to$ 40.50 in Nasdaq trading. <COREF ID = "30" TYPE = "IDENT" REF = "18"> Monday </COREF> 's enthusiasm among investors was in sharp contrast to the situation last fall, when <COREF ID = "31" TYPE = "IDENT" REF = "29"> Globalstar </COREF> was forced to withdraw a $ 400 million debt offering because of lack of interest. " <COREF ID = "32" TYPE = "IDENT" REF = "16" MIN = "deal"> This deal </COREF> means that <COREF ID = "33" TYPE = "IDENT" REF = "15"> Bernard Schwartz </COREF> can focus most of <COREF ID = "34" TYPE = "IDENT" REF = "33"> his </COREF> time

on <COREF ID = “35” TYPE = “IDENT” REF = “31”> Globalstar </COREF> and that is a key plus for <COREF ID = “36” TYPE = “IDENT” REF = “35”> Globalstar </COREF> because <COREF ID = “37” TYPE = “IDENT” REF = “34”> Bernard Schwartz </COREF> is brilliant, ” said <COREF ID = “39” MIN = “Robert Kaimowitz”> Robert Kaimowitz, <COREF ID = “38” TYPE = “IDENT” REF = “39” MIN = “analyst”> a satellite communications analyst at Unterberg Harris in New York </COREF> </COREF>. Though the <COREF ID = “44” MIN = “idea”> idea of setting up a global telephone network based on dozens of satellites </COREF> appears the stuff of science fiction, <COREF ID = “40” TYPE = “IDENT” REF = “37”> Schwartz </COREF> and many others, including Motorola Inc., several international telecommunications companies and <COREF ID = “42” MIN = “William Gates”> William Gates, the <COREF ID = “41” TYPE = “IDENT” REF = “42” MIN = “chairman”> chairman of Microsoft Corp. </COREF>, </COREF> see <COREF ID = “43” TYPE = “IDENT” REF = “44”> it </COREF> as <COREF ID = “45” TYPE = “IDENT” REF = “43” MIN = “opportunity”> a very real opportunity </COREF>. Already more than $ 3 billion has been raised for four competing projects. <COREF ID = “46” TYPE = “IDENT” REF = “40”> Schwartz </COREF> said <COREF ID = “47” TYPE = “IDENT” REF = “30”> Monday </COREF> that there were more than 3.9 billion people in the world without telephone service and 30 million people currently on waiting lists. If <COREF ID = “48” TYPE = “IDENT” REF = “36”> Globalstar </COREF> begins <COREF ID = “50” TYPE = “IDENT” REF = “51” MIN = “service”> <COREF ID = “49” TYPE = “IDENT” REF = “48”> its </COREF> service </COREF> on schedule in 1998, <COREF ID = “52” TYPE = “IDENT” REF = “46”> he </COREF> predicted that the <COREF ID = “53” TYPE = “IDENT” REF = “49”> company </COREF> would have 3 million customers by 2,002, bringing in <COREF ID = “55”> $ 2.7 billion </COREF> in <COREF ID = “54” TYPE = “IDENT” REF = “55” MIN = “revenue”> annual revenue </COREF>. In addition, <COREF ID = “56” TYPE = “IDENT” REF = “52”> Schwartz </COREF> said <COREF ID = “57” TYPE = “IDENT” REF = “23”> Loral Space </COREF> would use <COREF ID = “58” TYPE = “IDENT” REF = “57”> its </COREF> holdings in <COREF ID = “60” MIN = “Space Systems Loral”> Space Systems Loral, <COREF ID = “59” TYPE = “IDENT” REF = “60” MIN = “maker”> a private maker of satellites </COREF>, </COREF> to expand into the direct broadcast satellite business. “ Any service that is based on satellites is going to be a fertile area for <COREF ID = “61” TYPE = “IDENT” REF = “58”> our </COREF> investment, ” <COREF ID = “62” TYPE = “IDENT” REF = “56”>

he </COREF> said. Shares in <COREF ID = "63" TYPE = "IDENT" REF = "61"> Loral Space </COREF> will be distributed to <COREF ID = "64" TYPE = "IDENT" REF = "26"> Loral </COREF> shareholders. The <COREF ID = "65" TYPE = "IDENT" REF = "63" MIN = "company"> new company </COREF> will start life with no debt and $ 700 million in cash. <COREF ID = "66" TYPE = "IDENT" REF = "53"> Globalstar </COREF> still needs to raise <COREF ID = "70"> $ 600 million </COREF>, and <COREF ID = "67" TYPE = "IDENT" REF = "62"> Schwartz </COREF> said that the <COREF ID = "68" TYPE = "IDENT" REF = "66"> company </COREF> would try to raise the <COREF ID = "69" TYPE = "IDENT" REF = "70"> money </COREF> in the debt market. But if <COREF ID = "71" TYPE = "IDENT" REF = "68"> it </COREF> can not, <COREF ID = "76"> <COREF ID = "72" TYPE = "IDENT" REF = "65"> Loral Space </COREF> and <COREF ID = "73" TYPE = "IDENT" REF = "71"> Globalstar </COREF> 's 10 other partners </COREF> will put up the <COREF ID = "74" TYPE = "IDENT" REF = "69"> money </COREF> <COREF ID = "75" TYPE = "IDENT" REF = "76"> themselves </COREF>, <COREF ID = "77" TYPE = "IDENT" REF = "67"> he </COREF> said.
</DOC>

2. <DOC>
<DOCID> nyt960116.0264 </DOCID>
<COREF ID = "78" TYPE = "IDENT" REF = "79"> 01-08-96 </COREF>
<COREF ID = "1"> McDonald's <COREF ID = "3"> satellites </COREF>
</COREF>
<COREF ID = "6"> 01-16 </COREF>
<COREF ID = "0" TYPE = "IDENT" REF = "1"> McDonald's </COREF>
<COREF ID = "2" TYPE = "IDENT" REF = "3"> satellites </COREF>
Company spotlight : <COREF ID = "4" TYPE = "IDENT" REF = "0"> McDonald's </COREF> shops for <COREF ID = "25"> customers </COREF> at <COREF ID = "27"> Wal-Mart </COREF>
(For use by New York Times News Service clients)
By Shannon Stevens
c. 1996 <COREF ID = "8"> Bloomberg Business News </COREF>
<COREF ID = "13" MIN = "North Brunswick"> North Brunswick, New Jersey </COREF>, <COREF ID = "5" TYPE = "IDENT" REF = "6"> Jan. 16 </COREF> (<COREF ID = "7" TYPE = "IDENT" REF = "8"> Bloomberg </COREF>) <COREF ID = "10"> Todd Purvis </COREF> could n't see through the sleet and grime on <COREF ID = "9" TYPE = "IDENT" REF = "10"> his </COREF> windshield, so

<COREF ID = “11” TYPE = “IDENT” REF = “9”> he </COREF> stopped at a <COREF ID = “87”> Wal-Mart </COREF> in <COREF ID = “12” TYPE = “IDENT” REF = “13” MIN = “North Brunswick”> North Brunswick, New Jersey </COREF>, to buy <COREF ID = “20”> washer fluid </COREF>. Inside, <COREF ID = “14” TYPE = “IDENT” REF = “11”> he </COREF> saw those golden arches. In minutes , <COREF ID = “15” TYPE = “IDENT” REF = “14”> Purvis </COREF> was sitting in a <COREF ID = “92”> McDonald’s </COREF> wolfing down a cheeseburger and chugging a Coke. “ <COREF ID = “16” TYPE = “IDENT” REF = “15”> I </COREF> ’ m killing two birds with one stone, ” said the <COREF ID = “17” TYPE = “IDENT” REF = “16” MIN = “salesman”> 34-year-old construction-equipment salesman </COREF>. “ It ’ll take <COREF ID = “18” TYPE = “IDENT” REF = “17”> me </COREF> 10 minutes to eat lunch and buy <COREF ID = “19” TYPE = “IDENT” REF = “20”> washer fluid </COREF> and <COREF ID = “21” TYPE = “IDENT” REF = “18”> I </COREF> ’ m on <COREF ID = “22” TYPE = “IDENT” REF = “21”> my </COREF> way. ” <COREF ID = “23” TYPE = “IDENT” REF = “4”> McDonald’s Corp. </COREF> is shopping for <COREF ID = “24” TYPE = “IDENT” REF = “25”> customers </COREF> inside some of the <COREF ID = “41”> nation </COREF> ’s biggest retailers, including <COREF ID = “26” TYPE = “IDENT” REF = “27”> Wal-Mart Stores Inc. </COREF> and <COREF ID = “36”> Home Depot Inc. </COREF> And why not, since 75 percent of <COREF ID = “28” TYPE = “IDENT” REF = “23”> McDonald’s </COREF> diners decide to eat at <COREF ID = “29” TYPE = “IDENT” REF = “28”> its </COREF> restaurants less than five minutes in advance ? “ <COREF ID = “30” TYPE = “IDENT” REF = “29”> They </COREF> want to be the first sign you see when you get hungry, ” said <COREF ID = “32” MIN = “Dennis Lombardi”> Dennis Lombardi , <COREF ID = “31” TYPE = “IDENT” REF = “32” MIN = “analyst”> an analyst at <COREF ID = “157” MIN = “Technomics Inc.”> Chicago-based market researcher Technomics Inc. </COREF> </COREF> </COREF> <COREF ID = “33” TYPE = “IDENT” REF = “30”> McDonald’s </COREF> already has more than 1,000 so-called satellite restaurants inside <COREF ID = “34” TYPE = “IDENT” REF = “26”> Wal-Mart </COREF> and <COREF ID = “35” TYPE = “IDENT” REF = “36”> Home Depot </COREF> stores, as well as in <COREF ID = “120” MIN = “stations”> <COREF ID = “118”> Amoco Corp. </COREF> and <COREF ID = “122”> Chevron Corp. </COREF> filling stations </COREF>, airports, train stations and shopping malls. While still a small fraction of <COREF ID = “37” TYPE = “IDENT” REF = “33”> its </COREF> almost 18,000 restaurants, the <COREF ID

= “38” TYPE = “IDENT” REF = “2”> satellites </COREF> make up a rising portion of <COREF ID = “39” TYPE = “IDENT” REF = “37”> McDonald’s </COREF> new stores in the <COREF ID = “40” TYPE = “IDENT” REF = “41”> U.S. </COREF> “ <COREF ID = “42” TYPE = “IDENT” REF = “38”> Satellites </COREF> give us an opportunity to increase the number of customers we are able to satisfy with the <COREF ID = “43” TYPE = “IDENT” REF = “39”> McDonald’s </COREF> brand, ” said <COREF ID = “46” MIN = “Jack Greenberg”> <COREF ID = “45” TYPE = “IDENT” REF = “46” MIN = “Chief Financial Officer”> <COREF ID = “44” TYPE = “IDENT” REF = “43”> McDonald’s </COREF> Chief Financial Officer </COREF>, Jack Greenberg </COREF>. “ <COREF ID = “47” TYPE = “IDENT” REF = “42” STATUS = “OPT”> It </COREF> ’s a <COREF ID = “48” TYPE = “IDENT” REF = “47” STATUS = “OPT”> tool </COREF> in <COREF ID = “49” TYPE = “IDENT” REF = “44”> our </COREF> overall convenience strategy. ” <COREF ID = “50” TYPE = “IDENT” REF = “49” MIN = “McDonald’s”> Oak Brook, Illinois-based McDonald’s </COREF> opened 500 satellites <COREF ID = “133” MIN = “year”> last year </COREF> in the <COREF ID = “51” TYPE = “IDENT” REF = “40”> U.S. </COREF>, about half the restaurants <COREF ID = “52” TYPE = “IDENT” REF = “50”> it </COREF> opened in the <COREF ID = “53” TYPE = “IDENT” REF = “51”> country </COREF>. <COREF ID = “54” TYPE = “IDENT” REF = “52”> McDonald’s </COREF> plans to open 800 to 1,000 satellite restaurants this year, said <COREF ID = “56” MIN = “vice president”> Jim Johannesen, <COREF ID = “55” TYPE = “IDENT” REF = “56” MIN = “vice president”> vice president of site development for <COREF ID = “57” TYPE = “IDENT” REF = “54”> McDonald’s </COREF> </COREF> </COREF>. On top of generating growth, <COREF ID = “58” TYPE = “IDENT” REF = “42”> satellites </COREF> have another advantage : <COREF ID = “59” TYPE = “IDENT” REF = “58”> They </COREF> ’ re cheap, costing one-third the <COREF ID = “136”> $ 1.1 million </COREF> of a <COREF ID = “128” MIN = “McDonald’s”> free-standing McDonald’s </COREF>, analysts say. As a result, <COREF ID = “60” TYPE = “IDENT” REF = “59”> they </COREF> turn a profit quicker, <COREF ID = “61” TYPE = “IDENT” REF = “55”> Johannesen </COREF> said. <COREF ID = “62” TYPE = “IDENT” REF = “57”> McDonald’s </COREF> began testing <COREF ID = “63” TYPE = “IDENT” REF = “60” MIN = “restaurants”> satellite restaurants </COREF> in Wal-Marts in 1993 after several senior managers from <COREF ID = “64” TYPE = “IDENT” REF = “62”> McDonald’s </COREF> went to <COREF ID = “65” TYPE = “IDENT” REF = “34”> Wal-Mart

</COREF> 's Bentonville, Arkansas, headquarters to study <COREF ID = "66" TYPE = "IDENT" REF = "65"> its </COREF> operations. <COREF ID = "67" TYPE = "IDENT" REF = "66" MIN = "Wal-Mart"> Wal-Mart, <COREF ID = "68" TYPE = "IDENT" REF = "67" MIN = "retailer"> the <COREF ID = "69" TYPE = "IDENT" REF = "53"> country </COREF> 's biggest retailer with <COREF ID = "72" MIN = "stores"> 2,000 stores </COREF> </COREF>, </COREF> already had <COREF ID = "75"> restaurants </COREF> in most of <COREF ID = "71" TYPE = "IDENT" REF = "72" MIN = "stores"> <COREF ID = "70" TYPE = "IDENT" REF = "68"> its </COREF> stores </COREF>. <COREF ID = "73" TYPE = "IDENT" REF = "64"> McDonald's </COREF> agreed to take over some of <COREF ID = "74" TYPE = "IDENT" REF = "75"> them </COREF>. " It 's been good for both companies, " said <COREF ID = "78" MIN = "Buddy Burns"> Buddy Burns, <COREF ID = "77" TYPE = "IDENT" REF = "78" MIN = "manager"> <COREF ID = "76" TYPE = "IDENT" REF = "70"> Wal-Mart </COREF> 's manager of branded food service </COREF> </COREF>. " It adds to the overall shopping experience to have <COREF ID = "79" TYPE = "IDENT" REF = "73"> McDonald's </COREF> there. " That 's certainly how <COREF ID = "85" MIN = "Eileen Cook and her 22-month-old daughter"> <COREF ID = "81"> Eileen Cook </COREF> and <COREF ID = "83" MIN = "Jessie"> <COREF ID = "82" TYPE = "IDENT" REF = "83" MIN = "daughter"> <COREF ID = "80" TYPE = "IDENT" REF = "81"> her </COREF> 22-month-old daughter </COREF>, Jessie, </COREF> </COREF> see it. " When <COREF ID = "84" TYPE = "IDENT" REF = "85"> we </COREF> come to <COREF ID = "86" TYPE = "IDENT" REF = "87"> Wal-Mart </COREF> for <COREF ID = "103"> diapers </COREF>, <COREF ID = "88" TYPE = "IDENT" REF = "84"> we </COREF> come <COREF ID = "89" TYPE = "IDENT" REF = "90"> here </COREF>, " said <COREF ID = "91" TYPE = "IDENT" REF = "81"> Cook </COREF>, 31, sitting at a table in the <COREF ID = "90" TYPE = "IDENT" REF = "92"> McDonald's </COREF> inside the <COREF ID = "94" TYPE = "IDENT" REF = "86" MIN = "store"> <COREF ID = "93" TYPE = "IDENT" REF = "12" MIN = "North Brunswick"> North Brunswick, New Jersey </COREF>, store </COREF>. " <COREF ID = "95" TYPE = "IDENT" REF = "82"> She </COREF> loves <COREF ID = "99"> Chicken McNuggets </COREF>, " <COREF ID = "96" TYPE = "IDENT" REF = "91"> Cook </COREF> said, spurring Jessie to jump up and shout " <COREF ID = "97" TYPE = "IDENT" REF = "95"> I </COREF> love <COREF ID = "98" TYPE = "IDENT" REF = "99"> chicknuggets </COREF>. " <COREF ID = "100" TYPE = "IDENT" REF = "97">

Jessie </COREF> 's meal of <COREF ID = "101" TYPE = "IDENT" REF = "98"> Chicken McNuggets </COREF> and fries, though, only comes after a round of shopping for <COREF ID = "102" TYPE = "IDENT" REF = "103"> diapers </COREF> and other household goods. " <COREF ID = "105"> Mothers and kids </COREF> are <COREF ID = "104" TYPE = "IDENT" REF = "105" MIN = "customers"> <COREF ID = "106" TYPE = "IDENT" REF = "107"> our </COREF> No. 1 customers </COREF>, " <COREF ID = "108" TYPE = "IDENT" REF = "78" MIN = "Burns"> <COREF ID = "107" TYPE = "IDENT" REF = "76"> Wal-Mart </COREF> 's Burns </COREF> said of <COREF ID = "109" TYPE = "IDENT" REF = "79"> McDonald's </COREF>. Another favorite location for <COREF ID = "111" TYPE = "IDENT" REF = "63" MIN = "satellites"> <COREF ID = "110" TYPE = "IDENT" REF = "109"> McDonald's </COREF> satellites </COREF> is in <COREF ID = "112" TYPE = "IDENT" REF = "35" MIN = "Home Depot"> Home Depot, <COREF ID = "113" TYPE = "IDENT" REF = "112" MIN = "retailer"> the <COREF ID = "114" TYPE = "IDENT" REF = "69"> nation </COREF> 's largest home-improvement retailer with more than 400 stores </COREF> </COREF>. <COREF ID = "115" TYPE = "IDENT" REF = "110"> McDonald's </COREF> also is building <COREF ID = "116" TYPE = "IDENT" REF = "111" MIN = "restaurants"> satellite restaurants </COREF> in <COREF ID = "119" TYPE = "IDENT" REF = "120" MIN = "stations"> <COREF ID = "117" TYPE = "IDENT" REF = "118"> Amoco </COREF> and <COREF ID = "121" TYPE = "IDENT" REF = "122"> Chevron </COREF> gas stations </COREF>, providing <COREF ID = "124"> motorists </COREF> with a chance to fill <COREF ID = "123" TYPE = "IDENT" REF = "124"> their </COREF> cars and <COREF ID = "125" TYPE = "IDENT" REF = "123"> their </COREF> stomachs in one quick stop. Yet <COREF ID = "126" TYPE = "IDENT" REF = "116"> satellites </COREF> are no substitute for the <COREF ID = "127" TYPE = "IDENT" REF = "128" MIN = "McDonald's"> free-standing, full-size McDonald's </COREF>. <COREF ID = "130" TYPE = "IDENT" REF = "45" MIN = "Greenberg"> <COREF ID = "129" TYPE = "IDENT" REF = "115"> McDonald's </COREF> Greenberg </COREF> is quick to point out that the <COREF ID = "131" TYPE = "IDENT" REF = "129"> company </COREF> opened about 1,800 free-standing restaurants <COREF ID = "132" TYPE = "IDENT" REF = "133" MIN = "year"> last year </COREF>, 50 percent more than in 1994, partly the result of a cost-cutting program. <COREF ID = "134" TYPE = "IDENT" REF = "127" MIN = "restaurants"> Free-standing restaurants </COREF> now cost <COREF ID = "135" TYPE = "IDENT" REF = "136"> $ 1.1 million </COREF> each to build,

one-third less than a few years ago as the <COREF ID = “137” TYPE = “IDENT” REF = “131”> company </COREF> uses less expensive construction materials and designs that take less space. “ A few years ago <COREF ID = “138” TYPE = “IDENT” REF = “137”> we </COREF> were only opening a couple of hundred, ” <COREF ID = “139” TYPE = “IDENT” REF = “130”> Greenberg </COREF> said. “ With the low-cost approach, <COREF ID = “140” TYPE = “IDENT” REF = “138”> we </COREF> are able to open more. ” For now, <COREF ID = “141” TYPE = “IDENT” REF = “140”> McDonald’s </COREF> is n’t thinking about building many satellite restaurants outside the <COREF ID = “142” TYPE = “IDENT” REF = “114”> U.S. </COREF> In other countries, the <COREF ID = “143” TYPE = “IDENT” REF = “141”> company </COREF> is focused on building brand recognition and loyalty by opening <COREF ID = “144” TYPE = “IDENT” REF = “134” MIN = “restaurants”> traditional restaurants </COREF>, <COREF ID = “145” TYPE = “IDENT” REF = “139”> Greenberg </COREF> said. <COREF ID = “146” TYPE = “IDENT” REF = “143”> McDonald’s </COREF> is looking at joining up with other <COREF ID = “147” TYPE = “IDENT” REF = “142”> U.S. </COREF> retailers. <COREF ID = “148” TYPE = “IDENT” REF = “146”> It </COREF> recently opened a satellite restaurant in <COREF ID = “150” MIN = “Tandy ’s Incredible Universe”> <COREF ID = “149” TYPE = “IDENT” REF = “150” MIN = “superstore”> Tandy Corp. ’s new electronic superstore </COREF>, Tandy ’s Incredible Universe </COREF>. “ <COREF ID = “154”> You </COREF> have to pick <COREF ID = “151” TYPE = “IDENT” REF = “152” MIN = “partners”> <COREF ID = “153” TYPE = “IDENT” REF = “154”> your </COREF> partners </COREF> pretty carefully because <COREF ID = “152”> they </COREF> may not keep up to <COREF ID = “155” TYPE = “IDENT” REF = “153”> your </COREF> standards, ” said <COREF ID = “158” TYPE = “IDENT” REF = “31” MIN = “Lombardi”> <COREF ID = “156” TYPE = “IDENT” REF = “157”> Technomics </COREF> ’ Lombardi </COREF>. “ <COREF ID = “159” TYPE = “IDENT” REF = “148”> McDonald’s </COREF> is smart enough to be careful about that. ” Keep an eye out for <COREF ID = “161” MIN = “retailers”> hot new retailers </COREF> making <COREF ID = “160” TYPE = “IDENT” REF = “161”> their </COREF> way to the <COREF ID = “165”> top </COREF> do n’t be surprised if there ’s a <COREF ID = “162” TYPE = “IDENT” REF = “159”> McDonald’s </COREF> inside when <COREF ID = “163” TYPE = “IDENT” REF = “160”> it </COREF> gets <COREF ID = “164” TYPE = “IDENT” REF = “165”> there </COREF>.
</DOC>

Three MUC-7 documents for which a qualitative error analysis has been carried out

3. <DOC>
<DOCID> nyt960229.0649 </DOCID>
<COREF ID = “58” TYPE = “IDENT” REF = “2”> 02-26-96 </COREF>
<COREF ID = “1”> Silicon Graphics </COREF>
<COREF ID = “0” TYPE = “IDENT” REF = “1”> SGI </COREF>
making <COREF ID = “9” MIN = “games”> video games </COREF>
more lifelike
By Jeff Pelline
c . 1996 San Francisco Chronicle
<COREF ID = “2” TYPE = “IDENT” REF = “0” MIN = “company”>
The Silicon Valley company that helped create computer-generated dinosaurs in Jurassic Park </COREF> next week will unveil a new strategy to bring you <COREF ID = “7” MIN = “games”> more lifelike video games </COREF>. On <COREF ID = “18”> Monday </COREF>, <COREF ID = “24” MIN = “sources”> industry sources </COREF> said, <COREF ID = “3” TYPE = “IDENT” REF = “2” MIN = “Silicon Graphics Inc.”> Mountain View-based Silicon Graphics Inc. </COREF> will release a <COREF ID = “5” MIN = “technology”> technology dubbed <COREF ID = “4” TYPE = “IDENT” REF = “5” MIN = ““ FireWalker ””> “ FireWalker ” </COREF> </COREF> designed to make <COREF ID = “6” TYPE = “IDENT” REF = “7” MIN = “games”> the next generation of <COREF ID = “8” TYPE = “IDENT” REF = “9” MIN = “games”> video games </COREF> with 3-D images </COREF> more economical and commonplace . The <COREF ID = “10” TYPE = “IDENT” REF = “3”> company </COREF> ’s 112-year-old Silicon Studio subsidiary will work with Sega Enterprises of Japan, SegaSoft and Time Warner Interactive, among others, to test the <COREF ID = “11” TYPE = “IDENT” REF = “4”> software </COREF>. <COREF ID = “12” TYPE = “IDENT” REF = “11”> It </COREF> will be sold starting this summer. <COREF ID = “35”> San Francisco </COREF> ’s Rocket Science plans to release the first video game using the <COREF ID = “13” TYPE = “IDENT” REF = “12”> technology </COREF> by Christmas. <COREF ID = “14” TYPE = “IDENT” REF = “10”> Silicon Graphics </COREF> no doubt hopes <COREF ID = “15” TYPE = “IDENT” REF = “13”> “ FireWalker ” </COREF> will help jump start <COREF ID = “16” TYPE = “IDENT” REF = “14”> its </COREF> recent sluggish performance, but that ’s not guaranteed. Also on <COREF ID = “17” TYPE = “IDENT” REF = “18”> Monday </COREF>, the <COREF ID = “19” TYPE = “IDENT” REF = “16”> company </COREF> ’s high-tech <COREF ID = “49”> entertainment </COREF> group will announce <COREF ID = “32” MIN = “changes”> top management changes </COREF>. <COREF ID = “21” MIN = “Wei Yen and Eric Carlson”> <COREF ID = “20”

TYPE = “IDENT” REF = “21” MIN = “vice presidents”> Two key vice presidents </COREF>, Wei Yen and Eric Carlson, </COREF> are leaving to start <COREF ID = “22” TYPE = “IDENT” REF = “20”> their </COREF> own Silicon Valley companies, <COREF ID = “23” TYPE = “IDENT” REF = “24”> sources </COREF> said. <COREF ID = “26” MIN = “Rob Burgess”> <COREF ID = “25” TYPE = “IDENT” REF = “26” MIN = “executive”> Another executive from <COREF ID = “27” TYPE = “IDENT” REF = “19”> SGI </COREF> ’s Alias Wavefront subsidiary </COREF>, Rob Burgess, </COREF> may be reassigned to help take <COREF ID = “28” TYPE = “IDENT” REF = “22”> their </COREF> place. <COREF ID = “29” TYPE = “IDENT” REF = “27”> SGI </COREF> and the other companies involved declined comment. <COREF ID = “30” TYPE = “IDENT” REF = “29”> Silicon Graphics </COREF> chief executive Ed McCracken plans to announce the <COREF ID = “31” TYPE = “IDENT” REF = “32”> changes </COREF> on <COREF ID = “33” TYPE = “IDENT” REF = “17”> Monday </COREF> at a press conference in <COREF ID = “34” TYPE = “IDENT” REF = “35”> San Francisco </COREF>. <COREF ID = “37” TYPE = “IDENT” REF = “15” MIN = “technology”> <COREF ID = “36” TYPE = “IDENT” REF = “30”> SGI </COREF> ’s new <COREF ID = “38” TYPE = “IDENT” REF = “8” MIN = “game”> video-game </COREF> making technology </COREF> is akin to the high-tech breakthroughs that the <COREF ID = “39” TYPE = “IDENT” REF = “36” MIN = “maker”> computer workstation maker </COREF> brought to moviemaking in the late ’ 80s. The <COREF ID = “42” MIN = “products”> <COREF ID = “40” TYPE = “IDENT” REF = “39”> company </COREF> ’s products </COREF> now are <COREF ID = “41” TYPE = “IDENT” REF = “42” MIN = “equipment”> standard equipment </COREF> in <COREF ID = “46”> Hollywood </COREF>, along with computers from Sun Microsystems and Apple Computer. Future customers may include <COREF ID = “44” MIN = “DreamWorks SKG”> DreamWorks SKG, <COREF ID = “43” TYPE = “IDENT” REF = “44” MIN = “studio”> the <COREF ID = “45” TYPE = “IDENT” REF = “46”> Hollywood </COREF> studio that plans to create movies, <COREF ID = “47” TYPE = “IDENT” REF = “38” MIN = “games”> video games </COREF> and other interactive <COREF ID = “48” TYPE = “IDENT” REF = “49” STATUS = “OPT”> entertainment </COREF> </COREF> </COREF>. <COREF ID = “50” TYPE = “IDENT” REF = “43”> DreamWorks </COREF> and <COREF ID = “51” TYPE = “IDENT” REF = “40”> Silicon Graphics </COREF> already have forged a technology alliance. Why is <COREF ID = “52” TYPE = “IDENT” REF = “37”> “ FireWalker ” </COREF> so important ? Although the $ 10,000 price may sound hefty, the <COREF ID =

“53” TYPE = “IDENT” REF = “52”> product </COREF> is designed to make it cheaper, quicker and easier to make <COREF ID = “54” TYPE = “IDENT” REF = “6” MIN = “games”> video games that feature advanced special effects, such as lifelike body movements </COREF>. The <COREF ID = “55” TYPE = “IDENT” REF = “53”> technology </COREF> allows <COREF ID = “59” MIN = “makers”> <COREF ID = “56” TYPE = “IDENT” REF = “47” MIN = “game”> video game </COREF> makers </COREF> to automatically position characters in scenes without the need to program, saving time and money. <COREF ID = “58” TYPE = “IDENT” REF = “59” MIN = “makers”> <COREF ID = “57” TYPE = “IDENT” REF = “56” MIN = “game”> Video game </COREF> makers </COREF> are eager to cut <COREF ID = “60” TYPE = “IDENT” REF = “58”> their </COREF> costs because of the big expense, often more than $ 1 million over 11/2 years, required to produce a hit. The industry is undergoing a shakeout, largely because of the heavy capital investment required to survive. But for the winners, the payoff can be great. In <COREF ID = “62”> its </COREF> first week, <COREF ID = “61” TYPE = “IDENT” REF = “62” MIN = ““ Mortal Combat II ””> the video game “ Mortal Combat II ” </COREF> grossed $ 50 million, more than the movie “ Forrest Gump ” or “ Lion King. ” <COREF ID = “63” TYPE = “IDENT” REF = “48” STATUS = “OPT”> Entertainment </COREF> software titles are raking in more than <COREF ID = “65”> $ 3 billion </COREF> in <COREF ID = “64” TYPE = “IDENT” REF = “65”> sales </COREF> annually.
</DOC>

APPENDIX D

Three KNACK-2002 documents for which a qualitative error analysis has been carried out

This Appendix contains three texts from the KNACK-2002 test data. The texts are annotated with coreferential information according to the annotation guidelines given in Appendix A. And the manual error analysis in Chapter 8 is based on these three documents.

1. Document 1:

   <DOC>
   <DOCID> 09WWEEKQ_265.txt <DOCID>

   <COREF ID = “201”> Lionel Jospin <COREF> <COREF ID = “203” TYPE = “IDENT” REF = “201”> kandidaat <COREF>

   Met een eenvoudige fax naar het persbureau AFP stelt <COREF ID = “202” TYPE = “IDENT” REF = “201” MIN = “Lionel Jospin”> de Franse premier Lionel Jospin <COREF> zich officieel kandidaat voor de presidentsverkiezingen in <COREF ID = “205” TYPE = “IDENT” REF

= “201” > zijn <COREF> land. De eerste ronde daarvan wordt gehouden op 21 april. Een week eerder had <COREF ID = “208” MIN = “Jacques Chirac”> huidig president Jacques Chirac <COREF> <COREF ID = “209” TYPE = “IDENT” REF = “208”> zijn <COREF> kandidatuur met veel meer vlagvertoon bekendgemaakt. <COREF ID = “910” > < COREF ID = “909” TYPE = “IDENT” REF = “201” MIN = “Jospin” > De socialist Jospin <COREF> en <COREF ID = “211” TYPE = “IDENT” REF = “208” MIN = “Chirac”> de gaullist Chirac <COREF> <COREF> zijn <COREF ID = “911” TYPE = “IDENT” REF = “910” MIN = “kandidaten”> de belangrijkste kandidaten voor het hoogste ambt <COREF>. <COREF ID = “912” TYPE = “IDENT” REF = “910”>Ze <COREF> werken momenteel al bijna vijf jaar noodgedwongen samen. In de opiniepeilingen liggen <COREF ID = “913” TYPE = “IDENT” REF = “910”> <COREF ID = “215” TYPE = “IDENT” REF = “201”> Jospin <COREF > en <COREF ID = “216” TYPE = “IDENT” REF = “208”> Chirac <COREF> <COREF> zij aan zij.

<DOC>

2. Document 2:

<DOC>
<DOCID> 02BWEEKQ_206.txt <DOCID>

<COREF ID = “74”> Verkeersveiligheid <COREF>

<COREF ID = “76” MIN = “Steve Stevaert”> Vlaams minister van Mobiliteit Steve Stevaert <COREF> (SP.A) dreigt met een regeringscrisis als de federale regering blijft weigeren mee te werken aan <COREF ID = “87” MIN = “verbeteren”> het verbeteren van <COREF ID = “75” TYPE = “IDENT” REF = “74”> de verkeersveiligheid <COREF> <COREF>. Dat zegt <COREF ID = “77” TYPE = “IDENT” REF = “76”> hij <COREF > in het TV1-programma De Zevende Dag. < COREF ID = “78” TYPE = “IDENT” REF = “76” > Stevaert < COREF > ergert <COREF ID = “1” TYPE = “IDENT” REF = “78”> zich <COREF> aan de manier waarop <COREF ID = “89” MIN = “ministeries”> de verschillende ministeries <COREF> <COREF ID = “88” TYPE = “IDENT” REF = “87”> het dossier <COREF> naar <COREF ID = “90” TYPE = “IDENT” REF = “89”> elkaar <COREF> doorschuiven. Donderdag gaven <COREF ID = “92”> <COREF ID = “79” TYPE = “IDENT” REF = “76”> Stevaert <COREF> en <COREF ID =

“80”>Charles Picqué <COREF> <COREF> (PS), <COREF ID = “91” TYPE = “IDENT” REF = “80” MIN = “minister”> federaal minister van Economische Zaken <COREF>, <COREF ID = “93” TYPE = “IDENT” REF = “92”> elkaar <COREF> de schuld voor het disfunctioneren van <COREF ID = “83” MIN = “camera’s”> twee onbemande camera’s <COREF> op de A12 in Willebroek. <COREF ID = “653” TYPE = “IDENT” REF = “80”> Picqué <COREF > - bevoegd voor de erkenning van <COREF ID = “94” TYPE = “IDENT” REF = “83”> de flitspalen <COREF> - wilde <COREF ID = “84” TYPE = “IDENT” REF = “83” MIN = “camera’s”> de onbemande camera’s <COREF> niet homologeren omdat <COREF ID = “85” TYPE = “IDENT” REF = “83”> ze <COREF> ’ niet voldeden aan de opgelegde normen ’. <COREF ID = “652” TYPE = “IDENT” REF = “79”> Stevaert <COREF> - bevoegd voor het in orde brengen van <COREF ID = “86” TYPE = “IDENT” REF = “83” MIN = “camera’s”> de onbemande camera’s <COREF> - beweerde dat de aanpassingen al gebeurd zijn.

<DOC >

3. Document 3:

<DOC>
<DOCID> 02WWEEKQ_222.txt <DOCID>

<COREF ID = “206”>Twintigste kaper <COREF>

<COREF ID = “207”> Zacarias Moussaoui <COREF>, <COREF ID = “208” TYPE = “IDENT” REF = “207” MIN = “persoon”> de eerste persoon die door het Amerikaanse gerecht aangeklaagd is voor <COREF ID = “221” MIN = “terreuraanvallen”> de terreuraanvallen van 11 september <COREF> <COREF>, pleit onschuldig bij < COREF ID = “223” MIN = “verschijning”> < COREF ID = “209” TYPE = “IDENT” REF = “207”> zijn <COREF> eerste verschijning voor de rechtbank <COREF >. <COREF ID = “210” TYPE = “IDENT” REF = “207” MIN = “Fransman”> De Fransman van Marokkaanse afkomst <COREF> wordt ervan verdacht <COREF ID = “211” TYPE = “IDENT” REF = “206” MIN = “vliegtuigkaper”> de ‘ twintigste vliegtuigkaper ’ <COREF > te zijn die door omstandigheden (<COREF ID = “212” TYPE = “IDENT” REF = “207”> hij <COREF> zat in een Amerikaanse cel) niet aan <COREF ID = “222” TYPE = “IDENT” REF = “221”> de kapingen <COREF > kon deelnemen. <COREF ID = “215” MIN = “moeder”>

De moeder van <COREF ID = “213” TYPE = “IDENT” REF = “207”> Moussaoui <COREF> <COREF> vloog enige dagen voor <COREF ID = “224” TYPE = “IDENT” REF = “223”> < COREF ID = “214” TYPE = “IDENT” REF = “207” >zijn < COREF > voorleiding <COREF> naar de Verenigde Staten en gaf een persconferentie waarin <COREF ID = “216” TYPE = “IDENT” REF = “215”> ze <COREF> om een eerlijk proces voor <COREF ID = “218” TYPE = “IDENT” REF = “207” MIN = “zoon”> <COREF ID = “217” TYPE = “IDENT” REF = “215”> haar <COREF > zoon <COREF> vroeg. <COREF ID = “219” TYPE = “IDENT” REF = “207” > De beklaagde <COREF>, die de doodstraf riskeert, wil dat < COREF ID = “220” TYPE = “IDENT” REF = “219”> zijn <COREF> proces op televisie uitgezonden wordt.

<DOC>