COREFERENCE RESOLUTION: MAXIMUM METRIC SCORE TRAINING, DOMAIN ADAPTATION, AND ZERO PRONOUN RESOLUTION

SHANHENG ZHAO

NATIONAL UNIVERSITY OF SINGAPORE

2012

COREFERENCE RESOLUTION: MAXIMUM METRIC SCORE TRAINING, DOMAIN ADAPTATION, AND ZERO PRONOUN RESOLUTION

SHANHENG ZHAO
(B.E, SOUTH CHINA UNIVERSITY OF TECHNOLOGY)

A THESIS SUBMITTED
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTER SCIENCE
SCHOOL OF COMPUTING
NATIONAL UNIVERSITY OF SINGAPORE
2012

Acknowledgments

Writing this acknowledgement section reminds me of the last few days of my study at the National University of Singapore, the place where I spent the most valuable years of my life, the place which has enriched my academic learning and research experience, the place where I made many great friends.

Working on natural language processing in this thesis has been my main focus during the past few years. First of all, I would like to thank my advisor, Dr. Hwee Tou Ng, who led me all the way from day one. Not being familiar with natural language processing before enrolling in the doctorate program, I took much time to start from scratch. Dr. Ng exposed me to the world of statistical natural language processing. His profound insights on the field and penetrating advice helped me to achieve one milestone after another. Without his endless support, I would not have finished this thesis. I would like to take this opportunity to express my sincere gratitude to him for all that he has done for me.

I would also like to express my heartfelt gratitude and deepest respect to my thesis committee members, Dr. Chew Lim Tan and Dr. Min-Yen Kan. I met Dr. Tan even before coming to NUS. He is always very kind to me, willing to offer his endless help, both in work and in life. He is a truly respectable tutor. Dr. Min-Yen Kan is such a charismatic person who I can always learn something from in every conversation. When I asked him a question, no matter whether it is in a tea break between talks, during lunch time in the

canteen, or at numerous other places, he always answered it patiently and shed light on the problem.

My thanks also go to other faculty members in the School of Computing, NUS, who gave me great advice over the years: Dr. Wee Sun Lee and Dr. Tat-Seng Chua, as well as the research scientists from the Institute for Infocomm Research: Dr. Haizhou Li, Dr. Jian Su, and Dr. Min Zhang.

Among the most valuable memories I will take away from NUS are those of my great friends in the Computational Linguistics Lab: Yee Seng Chan, Tee Kiah Chia, Daniel Dahlmeier, Zheng Ping Jiang, Upali Kohomban, Ziheng Lin, Chang Liu, Jin Kiat Low, Wei Lu, Minh Thang Luong, Seung-Hoon Na, Preslav Nakov, Thanh Phong Pham, Long Qiu, Hendra Setiawan, Yee Fan Tan, Pidong Wang, Xuancong Wang, Hui Zhang, Jin Zhao, Zhi Zhong, Yu Zhou, and Muhua Zhu.

Though I am far away from home, my family is always there for me. My parents, my sister, my brother-in-law, and my newly-born niece are my strength to complete this thesis.

Finally, a big thank you goes to my fiancée Winnie, from the bottom of my heart, for her love and encouragement for so many years.

Contents

Summary

Coreference resolution is one of the central tasks in natural language processing. Successful coreference resolution benefits many other natural language processing and information extraction tasks. This thesis explores three important research issues in coreference resolution.

A large body of prior research on coreference resolution recasts the problem as a two-class classification problem. However, standard supervised machine learning algorithms that minimize classification errors on the training instances do not always lead to maximizing the F-measure of the chosen evaluation metric for coreference resolution. We propose a novel approach comprising the use of instance weighting and beam search to maximize the evaluation metric score on the training corpus during training. Experimental results show that this approach achieves significant improvement over the state of the art. We report results on standard benchmark corpora (two MUC corpora and three ACE corpora), when evaluated using the link-based MUC metric and the mention-based B-CUBED metric.

In the literature, most prior work on coreference resolution worked on newswire domain. Although a coreference resolution system trained on the newswire domain performs well on the same domain, there is a huge performance drop when it is applied to the biomedical domain. Annotating coreferential relations in a new domain is very time-consuming. This raises the question of how we can adapt a coreference resolution system trained on a

resource-rich domain to a new domain with minimum data annotations. We present an approach integrating domain adaptation with active learning to adapt coreference resolution from newswire domain to biomedical domain, and explore the effect of domain adaptation, active learning, and target domain instance weighting for coreference resolution. Experimental results show that domain adaptation with active learning and the weighting scheme achieves performance on MEDLINE abstracts similar to a system trained on full coreference annotation, but with a hugely reduced number of training instances that we need to annotate.

Lastly, we present a machine learning approach to the identification and resolution of Chinese anaphoric zero pronouns. We perform both identification and resolution automatically, with two sets of easily computable features. Experimental results show that our proposed learning approach achieves anaphoric zero pronoun resolution accuracy comparable to a previous state-of-the-art, heuristic rule-based approach. To our knowledge, our work is the first to perform both identification and resolution of Chinese anaphoric zero pronouns using a machine learning approach.

List of Tables

List of Figures

List of Algorithms

Chapter 1

Introduction

Natural language processing (NLP) is the field of using computers to manipulate human languages. It has a long history in the area of artificial intelligence (AI). Amongst many of the subtopics in natural language processing, coreference resolution is one of the most challenging.

In the early days of the literature, coreference resolution was studied mainly from a theoretical linguistics perspective. After the 1990s, the problem of coreference resolution has been subject to empirical evaluation. This thesis investigates the problems of maximizing coreference resolution metric score during training, domain adaptation in coreference resolution, as well as coreference resolution in non-English texts.

Coreference resolution is one of the core tasks in natural language processing. It is a key ingredient of discourse analysis. For example, coherence and information ordering analysis depend on accurate coreference resolution outputs (Barzilay and Lapata, 2005; Lapata and Barzilay, 2005). Successful coreference resolution also benefits other natural language processing tasks, such as information extraction (Kehler, 1997; Zelenko *et al.*, 2004), information retrieval (Na and Ng, 2009), question answering (Morton, 1999), text

summarization (Bergler *et al.*, 2003; Witte and Bergler, 2003; Steinberger *et al.*, 2005; Stoyanov and Cardie, 2006), and machine translation (Nakaiwa and Ikehara, 1992; He, 1998). Coreference resolution has become one of the standard steps in many of these tasks.

We start the chapter with the definition of coreference resolution. After that, we describe the motivations and contributions of the thesis. Finally, the outline of the thesis is given in Section 1.4.

1.1 Coreference Resolution

Coreference resolution refers to the process of determining whether two or more phrases refer to the same entity. In general, coreference resolution includes both intra-text (within the same text) resolution and inter-text (across text) resolution. In this thesis, we limit the scope to intra-text resolution, in other words, resolution of phrases within the same document.

Although most prior work on coreference resolution was on noun phrase (NP) coreference resolution, the research includes resolution of verb phrases, events, etc. However, we limit the scope of this thesis to noun phrase coreference resolution. In the remaining part of this thesis, if not stated, coreference resolution refers to intra-text noun phrase coreference resolution. The research on coreference resolution covers different languages. Some non-English languages have specific language phenomena which require extra efforts in coreference resolution, e.g., zero anaphora resolution in Chinese. In this thesis, we also investigate zero anaphora (which can be seen as a special noun phrase) resolution in Chinese.

1.1.1 Noun Phrase Coreference Resolution

Noun phrase coreference resolution, by definition, refers to the process of determining whether two or more noun phrases refer to the same entity in a discourse. A noun phrase can be a pronoun, common noun, or proper noun.

Here is an example:

> [*Bill Gates*]$_1$, [*the chairman*]$_2$ of [*Microsoft Corp.*]$_3$, announced [*his*]$_4$ retirement from [*the company*]$_5$.

In the above sentence, *Bill Gates*, *the chairman*, and *his* all refer to the same person and hence are coreferential, while *Microsoft Corp.* and *the company* both refer to the same company and hence are coreferential. All coreferential noun phrases referring to the same entity form a coreference chain. The task of coreference resolution is to determine these coreferential relations.

1.1.2 Anaphora Resolution

In most languages, there is a language phenomenon called reference: some texts cannot be interpreted semantically by their own, i.e., they make reference to something else for their interpretation. Halliday and Hasan (1976) categorized reference as exophora and endorphora.

Exophora, or exophoric reference, is reference to something that has not been explicitly encoded in the text. For example, *there* in *The chair over there is Tom's*.

Endophora, or endophoric reference, on the contrary, is reference to something within the text. Depending on where the referential expression is, endophora can be further categorized as anaphora and cataphora, which are references to the preceding text and to the following text, respectively.

Some linguists prefer to use the term anaphora to represent all of these referential effects. However, in this thesis, we follow the definitions in Halliday and Hasan (1976), in which anaphora is reference to the preceding text. The task of anaphora resolution is to determine the antecedent which interprets the anaphora.

Although there are subtle differences between coreference resolution and anaphora resolution (for example, see van Deemter and Kibble (2000)), we use the two terms interchangeably in this thesis, similar to most prior work in the literature.

1.1.3 Zero Pronoun Resolution

Every language has its own prominent language phenomena which make the language unique. Some of the phenomena in non-English languages bring extra challenges for coreference resolution in these languages compared to coreference resolution in English. One of these challenges is the prevalence of zero pronouns, which are very common in languages like Chinese, Japanese, Korean, Spanish, Italian, etc. In this thesis, we explore zero pronouns in Chinese.

A zero pronoun (ZP) is a gap (null element) in a sentence which refers to an entity that supplies the necessary information for interpreting the gap. In the literature, zero pronoun is also called ellipsis (Halliday and Hasan, 1976), or zero NP (Li, 2004).

A coreferential zero pronoun is a zero pronoun that is in a coreference relation to one or more overt noun phrases present in the same text. Here is an example of zero pronoun in Chinese from the Penn Chinese TreeBank (CTB) (Xue *et al.*, 2005) (sentence ID=300):

[中国	机电	产品	进出口	贸易]$_1$	继续	增加	,
[China	electronic	products	import and export	trade]$_1$	continues	increasing	,

ϕ_2	占	总	进出口	的	比重	继续	上升	。
ϕ_2	occupies	total	import and export	's	ratio	continues	increasing	.

where the anaphoric zero pronoun ϕ_2 refers to the noun phrase 中国机电产品进出口贸易.

Just like a coreferential noun phrase, a coreferential zero pronoun can also refer to a noun phrase in the preceding or the following text, called anaphoric zero pronoun (AZP) or cataphoric zero pronoun, respectively. Most coreferential zero pronouns in Chinese are anaphoric. In the corpus used in our evaluation, 98% of the coreferential zero pronouns have antecedents. Hence, for simplicity, we only consider anaphoric zero pronouns in this thesis. That is, we only attempt to resolve a coreferential zero pronoun to noun phrases preceding it.

Based on the above definition, the task of zero pronoun resolution is to resolve anaphoric zero pronouns to their correct antecedents. A typical zero pronoun resolution process comprises two stages. The first stage is the identification of the presence of the anaphoric zero pronouns. The second stage is resolving the identified anaphoric zero pronouns to the correct antecedents.

1.2 Motivation

Although the definition of coreference resolution is relatively simple, the task is considered a difficult natural language processing task. The resolution of coreferential noun phrases not only involves syntactic analysis, but also requires sophisticated semantic knowledge. The semantic knowledge can either be external world knowledge, or semantic knowledge

acquired from the text itself. In the literature of coreference resolution, syntactic, grammatical, and semantic features have been heavily exploited. Other knowledge sources and computational linguistics theories, e.g., semantic role labeling and centering theory, play an important role in coreference resolution as well. To solve the problem empirically, different machine learning approaches have been proposed for coreference resolution since the 1990s.

In the literature of research on coreference resolution, most prior work improves the performance of coreference resolution by exploiting fine-tuned feature sets and knowledge sources, or adopting alternative machine learning techniques and resolution methods during training and testing, respectively. However, most prior work ignores the fact that empirical risk minimization in standard supervised machine learning algorithms does not guarantee maximizing the F-measure of the chosen coreference evaluation metric. How to maximize the F-measure of the chosen coreference evaluation metric during training remains an open problem. Besides, most prior work on coreference resolution works on standard benchmark corpora in newswire domain in English. Relatively less prior research has explored other domains and languages, e.g., coreference resolution in biomedical texts or coreference resolution in Chinese. This motivates the need for exploring coreference resolution in non-newswire domain and non-English texts.

1.2.1 Maximum Metric Score Training

In the literature, most prior work on coreference resolution recasts the problem as a two-class classification problem. Machine learning-based classifiers are applied to determine whether a candidate anaphor and a potential antecedent are coreferential (Soon *et al.*, 2001; Ng and Cardie, 2002c; Stoyanov *et al.*, 2009).

Soon *et al.* (2001) introduced a machine learning framework for training and testing coreference resolution in general domain and reported performance comparable to the non-learning approaches. Under their framework, during training, a positive training instance is formed by a pair of markables, i.e., the anaphor and its closest antecedent. Each markable between the two, together with the anaphor, form a negative training instance. For example, in the sentence *"In a news release, the company said the new name more accurately reflects its focus on high-technology communications.*", the pair of *the company* and *its* forms a positive instance, while the pair of *the new name* and *its* forms a negative instance. A classifier is trained on all training instances by standard machine learning algorithms. During testing, all preceding markables of a candidate anaphor are considered as potential antecedents, and are tested in a back-to-front manner. The process stops if either an antecedent is found or the beginning of the text is reached.

Under this framework and its variants, a large body of prior research on coreference resolution follows the same process: during training, they apply standard supervised machine learning algorithms to minimize the number of misclassified training instances; during testing, they maximize either the local or the global probability of the coreferential relation assignments according to the specific chosen resolution method.

However, minimizing the number of misclassified training instances during training does not guarantee maximizing the score of the chosen evaluation metric for coreference resolution. First of all, coreference is a rare relation. There are far fewer positive training instances than negative ones. Simply minimizing the number of misclassified training instances is suboptimal and favors negative training instances. Second, evaluation metrics for coreference resolution are based on global assignments. Not all errors have the same impact on the metric score. Furthermore, the extracted training instances are not equally easy to be classified. In addition, if all pairs of noun phrase candidates are used during

training, data skewness is inevitable. If not all pairs of noun phrase candidates are used, it results in a loss of information. There is a trade-off between data skewness and loss of information.

Most of the work which follows the traditional training and resolution framework fails to recognize the fact that standard supervised learning algorithms that minimize classification errors over pair-wise training instances do not always lead to maximizing the F-measure of the chosen evaluation metric for coreference resolution.

1.2.2 Domain Adaptation for Coreference Resolution

A large body of prior research on coreference resolution focuses on texts in newswire domain. Standardized data sets, such as the MUC (DARPA Message Understanding Conference, (MUC-6, 1995; MUC-7, 1998)) and the ACE (NIST Automatic Content Extraction Entity Detection and Tracking task, (NIST, 2002)) data sets are widely used in the study of coreference resolution. There is a relatively small body of prior research on coreference resolution in non-newswire domain.

Traditionally, in order to apply supervised machine learning approaches to natural language processing problem in a specific domain, one needs to collect a text corpus in the domain and annotate training data. Annotating a data set in a new domain could be time-consuming and expensive. Comparing to other NLP tasks, e.g., part-of-speech (POS) tagging or named entity (NE) tagging, the annotation for coreference resolution is even more time-consuming and challenging. The reason is that in tasks like POS tagging, the annotator only needs to focus on the markable (a word, in the case of POS tagging) itself and a small window of neighbors. On the contrary, to annotate a coreferential relation, it takes the annotator much more effort. Traditionally, the annotator needs to first recognize whether a certain text span is a markable, and then scan through the text preceding the markable (a

potential anaphor) to look for potential antecedents. It also requires that the annotator understands the text to annotate the coreferential relation which is *semantic* in nature. If this markable is non-anaphoric, the annotator has to scan to the beginning of the text to know it. Furthermore, because coreferential relation is a pair-wise relation, the number of coreferential relations in a text is $O(n^2)$, where n is the number of markables in the text, compared to $O(n)$ in many other NLP tasks. This adds to the burden of data annotation in coreference resolution. Cohen *et al.* (2010) reported that it took an average of 20 hours to annotate coreferential relations on a single document with an average length of 6,155 words, while an annotator could annotate 3,000 words per hour in POS tag annotation (Marcus *et al.*, 1993).

It is time-consuming and expensive to annotate new data sets for new domains. The simplest approach to avoid this is to train a coreference resolution system on a resource-rich domain and apply it to a different target domain without any additional data annotation. Although coreference resolution systems work well on test texts in the same domain as the training texts, there is a huge performance drop when they are tested on a different domain, as illustrated by our experimental results reported in Chapter 4 of this thesis. This motivates the usage of domain adaptation techniques for coreference resolution: adapting or transferring a coreference resolution system from one source domain that we have a large collection of annotated data, to a second target domain in which we need good performance. It is almost inevitable to annotate *some* data in the target domain to achieve good coreference resolution performance. The question is how to minimize the amount of annotation needed. In the literature, active learning has been exploited to reduce the amount of annotation needed (Lewis and Gale, 1994). In contrast to annotating the entire data set, active learning queries only a subset of the data to annotate in an iterative process. Active learning is a less explored technique in the field of coreference resolution. Gasperin (2009)

tried to apply active learning for anaphora resolution, but found that using active learning was not better than randomly selecting the instances. How to apply active learning, especially integrating it with domain adaptation, remains an open problem for coreference resolution.

In recent years, with the advances in biology and life science research, there is a rapidly increasing number of biomedical texts, including research papers, patent documents, and the Web. This results in an increasing demand for applying natural language processing and information retrieval techniques to efficiently exploit text information in these large amounts of texts. Lately, biomedical text processing and mining has gained increasing attention and study in the community of NLP and IR, including not only biomedical text processing techniques that are biomedical domain dependent, but also domain adaptation techniques that adapt NLP/IR systems trained on other heavily studied and resource-rich domain to the biomedical domain with minimum data annotations. However, coreference resolution, one of the core tasks in natural language processing, has only a small body of prior research in the biomedical domain. The need of coreference resolution on biomedical texts and the small body of prior research make the biomedical domain a desirable target domain for evaluating domain adaptation for coreference resolution.

1.2.3 Zero Pronoun Resolution in Chinese

Much prior work on coreference resolution is on English texts. Relatively less work has been done on coreference resolution in other languages. At first glance, this is similar to domain adaptation: adapting a coreference resolution from English to another language. Many of the syntactic features for coreference resolution are language dependent, which makes a direct domain adaptation of coreference resolution from English to other languages relatively more challenging than domain adaptation of coreference resolution from

English	Portuguese	Italian	Chinese	Cantonese	Korean	Japanese
.96–.98	.56	.46–.56	.64	.58	.35–.45	.26–.38

Table 1.1: The percentages of the use of overt subjects in several languages collected by Kim (2000).

newswire domain to biomedical domain. Nevertheless, many principles and approaches for English coreference resolution are applicable to languages other than English.

However, there exist some language-specific linguistic phenomena which make coreference resolution in one language different from the others. Some of these phenomena in non-English languages bring extra challenges for coreference resolution in these languages compared to coreference resolution in English. One of these challenges is the prevalence of zero pronouns.

Zero pronouns occur much more frequently in Chinese than in English, and pose a unique challenge in coreference resolution for Chinese texts. For example, Kim (2000) conducted a study to compare the use of overt subjects in English, Chinese, and other languages (as shown in Table 1.1). He found that the use of overt subjects in English is over 96%, while this percentage is only 64% for Chinese, indicating that zero pronouns (lack of overt subjects) are much more prevalent in Chinese than in English.

In the literature, much of the work on coreference resolution is for English text. Furthermore, publicly available corpora for coreference resolution are mostly in English, e.g., the MUC and ACE data sets. Relatively less work has been done on coreference resolution for Chinese. Recently, the ACE Entity Detection and Tracking task included annotated Chinese corpora for coreference resolution. Florian *et al.* (2004), Zhou *et al.* (2005), and Wang and Ngai (2006) reported research on Chinese coreference resolution. However, they do not take into account zero pronouns, which is one of the major differences between coreference resolution in Chinese and coreference resolution in English.

Resolving an anaphoric zero pronoun to its correct antecedent in Chinese is a difficult

task. Although gender and number information is available for an overt pronoun and has proven to be useful in pronoun resolution in prior research, a zero pronoun in Chinese, unlike an overt pronoun, provides no such gender or number information. At the same time, identifying zero pronouns in Chinese is also a difficult task. There are only a few overt pronoun types in English, Chinese, and many other languages, and state-of-the-art part-of-speech taggers can successfully recognize most of these overt pronouns. However, zero pronouns in Chinese, which are not explicitly marked in a text, are hard to identify. Furthermore, even if a gap is a zero pronoun, it may not be coreferential. All these difficulties make the identification and resolution of anaphoric zero pronouns in Chinese a challenging task.

Chinese zero pronouns have been studied in linguistics research (Li and Thompson, 1979; Lee, 2002; Li, 2004), but only a small body of prior work in computational linguistics deals with Chinese zero pronoun identification and resolution (Yeh and Chen, 2004; Converse, 2006). To our knowledge, all previous research on zero pronoun identification and resolution in Chinese uses hand-engineered rules or heuristics. How to recast the task as a supervised machine learning problem and make use of the rapidly growing machine learning techniques to solve it remains an open problem.

1.3 Contributions of this Thesis

To address the issues described in Section 1.2, we propose a novel maximum metric score training (MMST) framework for coreference resolution. We explore domain adaptation for coreference resolution from newswire domain to biomedical domain. And we further explore coreference resolution in non-English texts, and propose the first machine learning-based zero pronoun identification and resolution system in Chinese. In this section, we

outline the work and the contributions of this thesis.

1.3.1 Maximum Metric Score Training

One of the conclusions which emerges particularly strongly from the review in the previous section is that minimizing the number of misclassified training instances during training, as most studies in the literature did, does not guarantee maximizing the F-measure of the chosen evaluation metric for coreference resolution during testing. Maximizing the evaluation metric score during testing, on the other hand, is time-consuming (Ng, 2005). Besides, the extracted training instances are not only not equally important, but also not equally easy to be classified. During testing, not all errors have the same impact on the evaluation metrics. Furthermore, it remains unclear what the best trade-off between data skewness and loss of information is.

In this thesis, the aim is to develop a novel approach comprising the use of instance weighting and beam search to address the issues above. Specifically, we propose a framework to

- provide an approach to maximizing the chosen evaluation metric score of coreference resolution on the training corpus during training;
- iteratively assign higher weights to the hard-to-classify training instances;
- utilize all pairs of noun phrase candidates during training to retain as much information as possible, as well as solve the data skewness problem automatically during training.

The approach proposed in this thesis closes a gap between training and testing a coreference resolution system that has not been addressed in the literature. The proposed maximum metric score training algorithm performs all metric score maximization during training, and outputs a standard classifier. Hence, during testing, it will be much faster than those approaches that optimize the assignment of coreferential relations during testing. It deepens the integration of coreference resolution and machine learning, and sheds light on the exploration of maximum metric score training on many other NLP tasks which traditionally train and test under different metrics.

In the study of maximum metric score training, we limit the scope to noun phrase coreference in English. However, the method is applicable to other languages. The input of the coreference resolution system is raw text. We do not assume any manually annotated information, e.g., part-of-speech tags, parse trees, or candidate markables. This results in a fully automatic system. Experimental results show that MMST achieves significant improvements over other baselines. Unlike most of the previous work, we report improved results over the state of the art on all five standard benchmark corpora (two MUC corpora and three ACE corpora), with both the link-based MUC metric and the mention-based B-CUBED metric.

1.3.2 Domain Adaptation for Coreference Resolution

In the previous section, we have pointed out that one of the most challenging obstacles in applying supervised learning approaches to coreference resolution is the difficulty of data annotation. It is much more time-consuming and expensive to annotate a corpus for coreference resolution than to annotate a corpus for other natural language processing tasks. Most existing annotated data sets for coreference resolution are in the newswire domain. To achieve good coreference resolution performance in a new domain, it is almost inevitable

to annotate some data. This raises the question of how to minimize the amount of data annotation needed while maintaining good coreference resolution performance. Although active learning has been successfully applied to other natural language processing and information retrieval tasks to reduce the amount of annotation needed, it remains an open problem how to apply the active learning technique to coreference resolution, especially integrating it with domain adaptation.

In this thesis, the aim is to explore domain adaptation for coreference resolution from a source domain for which we have a large collection of annotated data, to a second target domain that we want good performance, and to integrate domain adaptation with active learning to reduce the effort of data annotation in coreference resolution while maintaining comparable coreference resolution performance.

The approach proposed in this thesis comprises domain adaptation, active learning, and target domain instance weighting together to leverage the existing annotated corpora from newswire domain to reduce the cost of developing a coreference resolution system in biomedical domain. The approach achieves comparable coreference resolution performance on MEDLINE abstracts, but with a large reduction in the number of training instances that we need to annotate. To the best of our knowledge, our work is not only the first to use domain adaptation for coreference resolution, but also the first successful one to use active learning for coreference resolution.

In the study of domain adaptation for coreference resolution, we limit the scope to noun phrase coreference in English, and adapt from newswire domain to biomedical domain. However, the approach is generic and applicable to other domains. Again, the input of the coreference resolution system in both the source and the target domain is raw text.

1.3.3 Zero Pronoun Resolution in Chinese

In the previous section, we have pointed out that there exist some language-specific linguistic phenomena, e.g., the use of zero pronouns, which make coreference resolution in one language different from the others. Zero pronouns occur much more frequently in Chinese than in English, and pose a unique challenge to coreference resolution in Chinese. Although Chinese zero pronouns have been studied from the perspective of linguistics, only a small body of prior research studied this phenomenon from the perspective of computational linguistics. Furthermore, all previous research on zero pronoun identification and resolution in Chinese uses hand-engineered rules or heuristics. How to recast the task as a supervised machine learning problem and make use of the rapidly growing machine learning techniques to solve it remains an open problem.

In this thesis, we present a machine learning approach to the identification and resolution of Chinese anaphoric zero pronouns. We perform both identification and resolution automatically, with two sets of easily computable features. Experimental results show that our proposed learning approach achieves anaphoric zero pronoun resolution accuracy comparable to a previous state-of-the-art, heuristic rule-based approach. To our knowledge, our work is the first to perform both identification and resolution of Chinese anaphoric zero pronouns using a machine learning approach. Our proposed learning framework enables the application of rapidly growing machine learning techniques to further improve the performance of both the identification and resolution of Chinese anaphoric zero pronouns in the future.

In the study of Chinese zero pronouns, instead of conducting full coreference resolution for both noun phrases and zero pronouns, we focus on the task of anaphoric zero pronoun identification and resolution, as this is the major difference between coreference resolution

in Chinese and English. Most state-of-the-art approaches for noun phrase coreference resolution in English could be applied to noun phrase coreference resolution in Chinese, but not to anaphoric zero pronoun resolution in Chinese. Hence, we focus on this hard problem. Although our approach can be applied directly to machine-generated parse trees from raw text, in order to minimize errors introduced by preprocessing, and focus on the task itself, we use the gold standard word segmentation, POS tags, and parse trees provided by the Penn Chinese Treebank (CTB). However, we remove all null categories and functional tags from the CTB gold standard parse trees because null categories and functional tags are not typically present in the output of syntactic parsers, e.g., the Berkeley Parser.

1.4 Guide to the Thesis

In this chapter, we introduced the task of coreference resolution, outlined several research issues in coreference resolution, and gave an overview of the motivations and contributions of this thesis. The remaining part of this thesis is organized as follows. In Chapter 2, we first review the prior research on coreference resolution related to the research issues we address in the thesis. We then propose a maximum metric score training algorithm to maximize the chosen evaluation metric score for coreference resolution during training in Chapter 3. Chapter 4 describes the domain adaptation and active learning techniques for coreference resolution from newswire domain to biomedical domain. Chapter 5 presents the work on anaphoric zero pronoun identification and resolution for Chinese texts. Finally, we conclude the thesis and describe some potential future directions in Chapter 6.

Research carried out in this thesis has been presented in the following publications:

- **Shanheng Zhao** and Hwee Tou Ng. Identification and resolution of Chinese zero

pronouns: A machine learning approach. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL2007)*, pages 208–215, Prague, Czech Republic.

- **Shanheng Zhao** and Hwee Tou Ng. Maximum metric score training for coreference resolution. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING2010)*, pages 1308–1316, Beijing, China.

Chapter 2

Related Work

In Chapter 1, we have described the task of coreference resolution. The task has a long history in natural language processing, dating back to the 1960s. In the early days of the literature, coreference resolution was studied mainly from a linguistics perspective. Only after the 1990s, the problem of coreference resolution has been subject to empirical evaluation from the view of computational linguistics.

In this chapter, we first briefly review the history of coreference resolution from the perspective of computational linguistics in Section 2.1. Then we describe the work related to maximum metric score training for coreference resolution in Section 2.2. After that, we review the work related to domain adaptation for coreference resolution in Section 2.3. Finally, we discuss the work related to zero pronoun resolution in Chinese in Section 2.4.

2.1 A Brief Review for Coreference Resolution

Initially, coreference resolution, as a computational linguistics problem, was solved by rule-based approaches (Hobbs, 1978). Aone and Bennett (1995), Fisher *et al.* (1995), McCarthy and Lehnert (1995), McCarthy (1996), Kehler (1997), and Ge *et al.* (1998) are among

the first few works to address the problem based on learning from an annotated corpus. However, these approaches were either fine-tuned to a specific domain or did not perform as well as the rule-based systems.

Soon *et al.* (2001) introduced a general machine learning framework for training and testing coreference resolution and reported performance comparable to the non-learning approaches. Under their framework, during training, a positive training instance is formed by a pair of markables, i,e., the anaphor and its closest antecedent. Each markable between the two, together with the anaphor, form a negative training instance. A classifier is trained on all training instances by standard machine learning algorithms. During testing, all preceding markables of a candidate anaphor are considered as potential antecedents, and are tested in a back-to-front manner. The process stops if either an antecedent is found or the beginning of the text is reached. The motivation for the framework is straightforward. In both training and testing, it picks only the closest antecedent for an anaphor, given that coreference is a local linguistic phenomenon in many cases. The framework has been influential in the community of coreference resolution.

However, simply choosing the closest antecedent sometimes has a drawback. For example, in most situations, a pronoun is used as anaphora instead of cataphora. In the coreference chain *Clinton—he—the president*, it may not be appropriate to choose *he* as the antecedent of *the president*. There are no direct connections between the two NPs. They are coreferential to each other only because they are both coreferential to *Clinton*. Ng and Cardie (2002c) proposed a training and testing framework slightly modified from Soon *et al.* (2001): during training, the training instance selection for pronominal anaphor is the same as in Soon *et al.* (2001), but for non-pronominal anaphor, a positive training instance is formed by the anaphoric NP and its closest non-pronominal antecedent; during testing, the most probable preceding NP instead of the closest NP is selected as the antecedent.

Using this modified framework and an expanded feature set, their system achieves higher coreference resolution accuracies than Soon *et al.* (2001).

Following the pioneering work of Soon *et al.* (2001) and Ng and Cardie (2002c), recent work boosts the performance of coreference resolution by exploiting fine-tuned feature sets, focusing on particular issues of coreference, or adopting alternative resolution methods during testing. For example, Ng and Cardie (2002c), Versley *et al.* (2008a), and Versley *et al.* (2008b) employed expanded feature sets. Yang *et al.* (2004b), and Bergsma and Lin (2006) focused on the problem of pronoun resolution, while Gasperin and Vieira (2004), Poesio *et al.* (2004), and Vieira *et al.* (2006) investigated bridging reference. Ng and Cardie (2002b) and Ng (2004b) pointed out that identifying the anaphoricity of an NP before resolving it significantly improves the performance of coreference resolution.

Although these approaches gain improvement on the performance of coreference resolution, they all treat coreferential relation locally, i.e., between two NPs, and simply cluster all coreferential pairs of NPs together during testing, leaving the information provided by neighboring NPs unconsumed in both training and testing. Hence, they lack the ability of providing a global picture on how the coreference chains are formed. Yang *et al.* (2003) proposed a twin candidate model, in which during testing, all preceding NPs of a potential anaphor are competing against each other as an antecedent. Their approach differs from the traditional framework by not creating training and testing instance as a pair of two NPs, but a group of three. This is a kind of approximations of global inference. Denis and Baldridge (2007), on the other hand, proposed the use of integer linear programming (ILP) to jointly determine anaphoricity and antecedents, and maximize the global probability of coreferential relation assignments during testing. Finkel and Manning (2008) improved the ILP method by enforcing transitivity.

In recent years, exploiting semantic knowledge for coreference resolution has gained

attention in the community (Ng, 2007; Haghighi and Klein, 2010). Semantics is playing an increasingly important role in coreference resolution. Successful exploitation of semantic knowledge has the potential to boost the performance of coreference resolution to the next level. Other works on exploiting semantic knowledge for coreference resolution include the use of WordNet (Harabagiu *et al.*, 2001), Wikipedia (Ponzetto and Strube, 2006), semantic role labeling (Kong *et al.*, 2009), as well as various semantic patterns (Haghighi and Klein, 2009).

2.2 Maximum Metric Score Training

Most of the aforementioned work follows the same process: during training, they apply standard supervised machine learning algorithms to minimize the number of misclassified training instances; during testing, they maximize either the local or the global probability of the coreferential relation assignments according to the specific chosen resolution method. However, minimizing classification errors during training does not guarantee maximizing the F-measure of the chosen coreference evaluation metric.

Ng (2005) proposed a ranking model to maximize F-measure during testing. In the approach, n different coreference outputs for each test text are generated, by varying four components in a coreference resolution system, i.e., the learning algorithm, the instance creation method, the feature set, and the clustering algorithm. An SVM-based ranker then picks the output that is likely to have the highest F-measure. However, this approach is very time-consuming during testing, as F-measure maximization is performed during testing. This limits its usage on very large corpora.

In the community of machine learning, researchers have proposed approaches for learning a model to optimize a chosen evaluation metric other than classification accuracy on

all training instances. Joachims (2005) suggested the use of support vector machines to optimize nonlinear evaluation metrics. Cost sensitive learning has also been explored in machine learning research (Domingos, 1999; Elkan, 2001; Zadrozny and Elkan, 2001; Zadrozny *et al.*, 2003). However, these approaches do not differentiate between the errors in the same category in the contingency table. Furthermore, they do not take into account inter-instance relation (e.g., transitivity), which the evaluation metric for coreference resolution cares about.

Daume III (2006) proposed a structured learning framework for coreference resolution to approximately optimize the ACE metric. Our proposed approach differs in two aspects. First, we directly optimize the evaluation metric itself, and not by approximation. Second, unlike the incremental local loss in Daume III (2006), we evaluate the metric score globally.

In contrast to Ng (2005), Ng and Cardie (2002a) proposed a rule-induction system that maximizes the F-measure with rule-pruning during training. However, their approach is specific to rule induction, and is not applicable to other supervised learning classifiers. Ng (2004a) varied different components of coreference resolution, choosing the combination of components that results in a classifier with the highest F-measure on a held-out development set during training. In contrast, our proposed approach employs instance weighting and beam search to maximize the F-measure of the evaluation metric during training. Our approach is general and applicable to any supervised learning classifiers.

Recently, Wick and McCallum (2009) proposed a partition-wise model for coreference resolution to maximize a chosen evaluation metric using the Metropolis-Hastings algorithm (Metropolis *et al.*, 1953; Hastings, 1970). However, they found that in most cases, training on classification accuracy outperformed training on the coreference evaluation metrics. Furthermore, similar to Ng (2005), their approach requires the generation of multiple coreference assignments during testing.

Vemulapalli *et al.* (2009) proposed a document-level boosting technique for coreference resolution by re-weighting the documents that have the lowest F-measures. By combining multiple classifiers generated in multiple iterations, they achieved a CEAF (Constrained Entity-Alignment F-Measure) score slightly better than the baseline. Different from them, our approach works at the instance level, and we output a single classifier.

In the community of natural language processing, learning to optimize a particular metric score has been studied, e.g., the well-known minimum error rate training (MERT) in statistical machine translation (Och, 2003). Different from machine translation, coreference resolution requires more investigations into inter-instance relations because of the transitivity property of coreference resolution.

2.3 Domain Adaptation for Coreference Resolution

Most prior studies on coreference resolution worked on the newswire domain. Not only is there a relatively small body of prior research on coreference resolution in the biomedical domain, there are also fewer annotated corpora in this domain. Castaño *et al.* (2002) are among the first to annotate coreferential relations on biomedical domain. Their annotation only concerned the pronominal and nominal anaphoric expressions in 46 biomedical abstracts. Gasperin *et al.* (2007) annotated coreferential relations on 5 full articles in the biomedical domain, but only on noun phrases referring to bio-entities. Yang *et al.* (2004a) and Yang *et al.* (2004c) annotated full NP coreferential relations on biomedical abstracts of the GENIA corpus. The ongoing project of the CRAFT corpus is expected to annotate all coreferential relations on full text of biomedical articles (Cohen *et al.*, 2010).

Unlike the work of Castaño *et al.* (2002), Gasperin and Briscoe (2008), and Gasperin

(2009) that resolved coreferential relations on certain restricted entities in biomedical domain, we resolve full NP coreferential relations in the biomedical domain. Although the GENIA corpus contains 1,999 biomedical abstracts, Yang *et al.* (2004a) and Yang *et al.* (2004c) experimented on a subset of 200 and 100 abstracts, respectively. Yang *et al.* (2004c) reported an F-measure of coreference resolution of 81.7% on 30 test abstracts. Yang *et al.* (2004a) reported an F-measure of 68.3% under 5-fold cross validation on the 200 abstracts. In contrast, we randomly selected 399 abstracts in the 1,999 abstracts of the GENIA corpus as the test set, which is much larger than the previous work.

Domain adaptation has been studied and successfully applied to many natural language processing tasks (Jiang and Zhai, 2007; Daume III, 2007; Dahlmeier and Ng, 2010). On the other hand, active learning has also been applied to NLP tasks to reduce the need of data annotation in the literature (Tang *et al.*, 2002; Zhu *et al.*, 2010). Unlike the aforementioned work that applied one of domain adaptation or active learning to NLP tasks, we combine both. There is relatively less research on combining domain adaptation and active learning together for NLP tasks (Chan and Ng, 2007; Zhong *et al.*, 2008; Rai *et al.*, 2010). Chan and Ng (2007) and Zhong *et al.* (2008) used *count merging* and *augment*, respectively, as their domain adaptation techniques whereas we apply and compare multiple state-of-the-art domain adaptation techniques. Rai *et al.* (2010) exploited a streaming active learning setting whereas ours is pool-based. Dahlmeier and Ng (2010) evaluated the performance of three recently proposed domain adaptation algorithms for semantic role labeling. They evaluated the performance of domain adaptation with different sizes of target domain training data. In each of their experiments with a certain target domain training data size, the target domain training data were added all at once. In contrast, we add the target domain training instances selectively and iteratively. Compared to Dahlmeier and Ng (2010), we give weight to the target domain instances to further boost the performance

of domain adaptation. Most important, we work on coreference resolution, and this is the first systematic study of domain adaptation with active learning for coreference resolution. Although Gasperin (2009) tried to apply active learning for anaphora resolution, her results were negative: using active learning was not better than randomly selecting the instances.

2.4 Zero Pronoun Resolution in Chinese

Zero pronoun identification and resolution have a relatively small body of prior work. Much prior work on Chinese zero pronouns is from the view of linguistics or psycholinguistics (Li and Thompson, 1979; He, 1998; Huang, 1992). Besides Kim (2000), Tao and Healy (2005) also found that Chinese makes greater use of zero pronouns than English, and native Chinese speakers are better than native English speakers in interpreting zero pronouns.

From the perspective of computational linguistics, zero pronoun resolution in Chinese were resolved in a rule-based manner. Converse (2006) assumed that the gold standard Chinese anaphoric zero pronouns and the gold standard parse trees of the texts in Penn Chinese TreeBank (CTB) were given as input to her system, which resolved anaphoric zero pronouns using the Hobbs algorithm (Hobbs, 1978). Her system did not identify the anaphoric zero pronouns automatically. Yeh and Chen (2004) proposed an approach for Chinese zero pronoun resolution based on the Centering Theory (Grosz *et al.*, 1995). Their system used a set of hand-engineered rules to perform zero pronoun identification, and resolved zero pronouns with a set of hand-engineered resolution rules.

In languages other than Chinese, hand-engineered rule-based approaches were also applied to zero pronoun identification and resolution. For example, Ferrández and Peral (2000) proposed a hand-engineered rule-based approach to both identifying and resolving zero pronouns that are in the subject grammatical position in Spanish.

Besides hand-engineered rule-based approaches, applying machine learning approaches to zero pronoun resolution has also been studied in the literature. Iida *et al.* (2006) proposed a machine learning approach to resolving zero pronouns in Japanese using syntactic patterns. Their system also did not perform zero pronoun identification, and assumed that correctly identified zero pronouns were given as input to their system. The probabilistic model of Seki *et al.* (2002) both identified and resolved Japanese zero pronouns, with the help of a verb dictionary. Their model needed large-scale corpora to estimate the probabilities and to prevent data sparseness. Other works on Japanese zero pronoun resolution include Nakaiwa and Ikehara (1992), Nakaiwa and Shirai (1996), Okumura and Tamura (1996), and Kawahara and Kurohashi (2004).

To our knowledge, all previous research on zero pronoun identification and resolution in Chinese uses hand-engineered rules or heuristics, and our work is the first to perform both identification and resolution of Chinese anaphoric zero pronouns using a machine learning approach.

2.5 Summary

In this chapter, we have reviewed the brief history of coreference resolution, the work related to maximum metric score training for coreference resolution, domain adaptation for coreference resolution, as well as zero pronoun resolution in Chinese.

Chapter 3

Maximum Metric Score Training

In Chapter 1 and Chapter 2, we have shown that most prior work on coreference resolution recasts the task as a two-class classification problem and applies machine learning-based classifiers to determine whether a candidate anaphor and a potential antecedent are coreferential. However, minimizing the number of misclassified training instances during training, as most studies in the literature did, does not guarantee maximizing the F-measure of the chosen evaluation metric for coreference resolution during testing. Maximizing the evaluation metric score during testing, on the other hand, is time-consuming. Besides, the extracted training instances are not only not equally important, but also not equally easy to be classified. Furthermore, it remains unclear what the best trade-off between data skewness and loss of information is.

In this chapter, we describe a novel approach comprising the use of instance weighting and beam search to address the above issues. Our proposed maximum metric score training algorithm maximizes the chosen evaluation metric score on the training corpus during training. It iteratively assigns higher weights to the hard-to-classify training instances. The output of training is a standard classifier. Hence, during testing, MMST is faster than the

approaches which optimize the assignment of coreferential relations during testing. Experimental results show that MMST achieves significant improvements over the baselines. Unlike most of the previous work, we report improved results over the state of the art on all five standard benchmark corpora (two MUC corpora and three ACE corpora), with both the link-based MUC metric and the mention-based B-CUBED metric.

Since our approach aims to maximize the evaluation metric for coreference resolution, we start the chapter by first introducing the evaluation metrics for coreference resolution in Section 3.1. Next, Section 3.2 describes the training and resolution framework for coreference resolution. After that, we propose the novel maximum metric score training algorithm in Section 3.3. Experimental settings and results are presented in Section 3.4. Finally, we conclude the chapter in Section 3.5.

3.1 Evaluation Metrics

In the literature, various evaluation metrics have been proposed for coreference resolution, including the MUC metric (Vilain *et al.*, 1995), the B-CUBED metric (Bagga and Baldwin, 1998), the ACE metric (NIST, 2002), and the CEAF metric (Luo, 2005). Besides the evaluation metrics specifically designed for coreference resolution, resolution accuracies of one or more particular NP types are also reported in some prior work. Among all evaluation metrics, the MUC and the B-CUBED metrics are the most widely used in the literature. In this chapter, similar to most prior work, we report results in the MUC and the B-CUBED metrics.

The terminology used in the community of coreference resolution is mixed. Different corpora or papers use different sets of terminology. Before getting into the details of the evaluation metrics, we first introduce the terminology we will use throughout this thesis.

We define the following terms:

- Key: the coreference chains in the manually annotated gold standard.
- Response: the coreference chains output by a coreference resolution system.
- Markable or mention: a noun phrase which satisfies the markable definition in an individual corpus.
- Coreference chain: a cluster, or an equivalence class, formed by a set of coreferential markables in the key or the response.
- Link: a pair of coreferential markables.
- Singleton: a markable not coreferential to any other markables (in other words, it does not belong to any links).

Both the MUC and the B-CUBED metrics compare the coreference chains in the key and the response to compute the metric score. Like many other evaluation metrics for natural language processing and information retrieval, these two metrics compute the scores in terms of recall, precision, and F-measure.

Generally, recall measures how much relevant information the system has extracted from the text, while precision measures how much of the information that the system returned is actually correct (Jurafsky and Martin, 2000). F-measure combines recall and precision, and it is computed as:

$$F = \frac{(1 + \beta^2) \times Recall \times Precision}{Recall + \beta^2 \times Precision} \tag{3.1}$$

where β controls the importance of recall and precision. If $\beta = 1$, it will give equal importance to recall and precision, and the F-measure becomes the F_1-measure:

$$F_1 = \frac{2 \cdot Recall \cdot Precision}{Recall + Precision}$$

In the rest of the thesis, if not stated, for simplicity, we use F-measure to represent F_1-measure.

3.1.1 The MUC Evaluation Metric

Vilain *et al.* (1995) introduced the link-based MUC evaluation metric for the MUC-6 and MUC-7 coreference tasks.

Let

- S_i be the i-th coreference chain, or equivalence class, in the key.
- R_j be the j-th coreference chain, or equivalence class, in the response.
- $p(S_i)$ be a partition of S_i relative to the response.
- $p(R_j)$ be a partition of R_j relative to the key.

Recall is the number of correctly identified links over the number of links in the key:

$$Recall = \frac{\sum(|S_i| - |p(S_i)|)}{\sum(|S_i| - 1)}$$

Precision, on the other hand, is defined in the opposite way by switching the role of key and response:

$$Precision = \frac{\sum(|R_j| - |p(R_j)|)}{\sum(|R_j| - 1)}$$

F-measure is the trade-off between recall and precision as described above.

3.1.2 The B-CUBED Evaluation Metric

Bagga and Baldwin (1998) introduced the mention-based B-CUBED metric. The B-CUBED metric measures the accuracy of coreference resolution based on individual markables. Hence, it also gives credit to the identification of singletons, which the MUC metric does not.

In the B-CUBED metric, recall is computed as

$$Recall = \frac{1}{N} \sum_{d \in D} \sum_{m \in d} \frac{|O_m|}{|S_m|}$$

where D, d, and m are the set of documents, a document, and a markable, respectively. S_m is the coreference chain generated by the key that contains m, while O_m is the overlap of S_m and the coreference chain generated by the response that contains m. N is the total number of markables in D.

The precision, again, is computed by switching the role of key and response. Like the MUC metric, F-measure for the B-CUBED metric is the trade-off between recall and precision as described above.

3.2 The Coreference Resolution Framework

In Chapter 1, we have briefly described the coreference resolution framework introduced by Soon *et al.* (2001). In this section, we will describe the framework and its variants in details.

Algorithm 3.1 and 3.2 show the general training and resolution (testing) framework for coreference resolution. Typically, in supervised learning-based coreference resolution, the method first extracts a set of markables from each text, then trains a model on them or resolves them. The definition of markables is corpus-dependent.

Algorithm 3.1 A general training framework for coreference resolution

INPUT: A set of annotated texts T_1, T_2, . . . , T_n
OUTPUT: A classifier C

for all i in 1, 2, . . ., n **do**
  extract markables in T_i
  for each markable m in T_i **do**
    extract training instances, with m being the 2nd markable in the pair
  end for
end for
train a classifier C using the extracted training instances
return classifier C

3.2.1 Training

In coreference training, a training instance is a pair of different markables in the same text. Different instance selection strategies have been proposed in the literature. The most widely used strategy is the one proposed by Soon *et al.* (2001): a positive training instance (m_i, m_j) ($i < j$) is formed by an anaphoric markable m_j and its closest antecedent m_i. Each markable between the two, together with the anaphor, form a negative training instance: (m_{i+1}, m_j), (m_{i+2}, m_j), . . ., (m_{j-1}, m_j).

Ng and Cardie (2002c) proposed a training framework slightly modified from Soon *et al.* (2001): the training instance selection for a pronominal anaphor is the same as in Soon *et al.* (2001), but for a non-pronominal anaphor, a positive training instance is formed by an anaphoric markable and its closest non-pronominal antecedent.

Both Soon *et al.* (2001) and Ng and Cardie (2002c) sample only a small subset of all possible markable pairs. A large portion of information provided by all possible pairs is lost. To keep as much information as possible, one strategy is to use all possible markable pairs as training instances, in which a training instance is positive if the two markables are coreferential, and negative if they are not (McCarthy and Lehnert, 1995; Stoyanov *et al.*,

2009).

However, coreference is a rare relation with far fewer positive training instances than negative ones. If all pairs of markables are used during training, data skewness is inevitable. If not all pairs of markables are used, it results in a loss of information. It remains unclear what the best trade-off between data skewness and loss of information is.

Furthermore, some coreferential markable pairs do not have direct connections between the two markables. Including these pairs with positive class labels in the training data may lead to an incorrect classifier. For example, in a coreference chain *President Clinton–he–the president* (appearing in the order of their locations in the text), without looking at other markables in the coreference chain, there might not be sufficient information and knowledge to tell that *the president* is coreferential to *he*. They are coreferential because they both are coreferential to *President Clinton*.

After extracting the training instances, a classifier is trained on all training instances by standard machine learning algorithms.

3.2.2 Resolution

To determine the coreference relations of a given text, a test instance is a pair of markables in the text. Similar to training, different resolution strategies have also been proposed in the literature. The most widely used strategy is the one proposed by Soon *et al.* (2001): all preceding markables of a candidate anaphor are considered as potential antecedents, and are tested in a back-to-front manner. In other words, we test (m_{j-1}, m_j), (m_{j-2}, m_j), ..., sequentially, to look for the antecedent of m_j. If a pair of markables (m_i, m_j) is classified as positive by the classifier, i.e., the two markables are predicted to be coreferential, the first markable m_i is chosen as the antecedent of m_j. The process stops if either an antecedent is found or the beginning of the text is reached.

Algorithm 3.2 A general resolution framework for coreference resolution

INPUT: An unannotated text T
A classifier C
OUTPUT: T with coreferential annotations

extract markables in T
for each markable m in T **do**
  extract testing instances, with m being the 2nd markable in the pair
  classify all testing instances with C
  select one or more antecedents for m based on the classification decisions
end for
cluster the coreferential links to form coreferential chains
return T with the annotated coreferential chains

Ng and Cardie (2002c) used a slightly different resolution strategy: the most probable preceding markable instead of the closest one is selected as the antecedent.

Unlike Soon *et al.* (2001) and Ng and Cardie (2002c) which select only one antecedent for each anaphoric markable, McCarthy and Lehnert (1995) and Stoyanov *et al.* (2009) resolve an anaphoric markable to all coreferential antecedents.

After selecting the links for each markable, a clustering algorithm will group them together to form coreference chains: the individual classification decisions made by the coreference classifier do not guarantee that transitivity of coreferential NPs is obeyed. So it can happen that the pair A and B, and the pair B and C are both classified as coreferential, but the pair A and C is not classified as coreferential by the classifier. After all coreferential markable pairs are found (no matter by closest-first, best-first, or resolving-all strategies as in different prior work), all coreferential pairs are clustered together to form the coreference output. By doing so, transitivity is kept: a markable is in a coreference chain if and only if it is classified to be coreferential to at least one other markable in the chain. In the above example, the markables A, B, and C will be clustered together into the same coreference chain, even though the pair A and C is not classified as coreferential by the classifier.

3.3 Maximum Metric Score Training

In this section, we describe the proposed maximum metric score training algorithm. As discussed in the previous section, if not all pairs of markables are selected as training instances, there is information loss. In the MMST algorithm, we use all pairs of markables as training instances to keep as much information as possible.

In Chapter 1 and the previous section, we have also noted that not all extracted instances are equally important or are equally easy to be classified, thus treating them in the same way might be sub-optimal. Our MMST algorithm comprises the use of beam search and instance weighting to solve the problems. Initially all the pairs are equally weighted. We then iteratively assign more weights to the hard-to-classify pairs. The iterative process is conducted by a beam search algorithm.

3.3.1 Instance Weighting

Supervised learning algorithms are not perfect. During testing, they make errors. If an instance is positive in the gold standard but predicted as negative by the classifier, it is called *false negative*; if an instance is negative in the gold standard but predicted as positive by the classifier, it is called *false positive* (Russell and Norvig, 2002). Different from many other problems, in coreference resolution, these two types of errors have different impacts on forming the coreference chains, and hence the evaluation metric score.

The motivation of the approach is simple. Because during testing, clustering will impose and guarantee transitivity amongst a coreference chain, we do not need to (and as discussed, in many cases it is almost impossible to) find all positive pairs (links).

If we treat each markable as a vertex in a graph, and there is an edge between two coreferential markables, a coreference chain then forms a fully connected graph. According

to spanning tree theory, to connect n vertices together, the least number of edges we need is $n-1$ (Cormen *et al.*, 2001). If we can find $n-1$ edges to form a spanning tree for these n vertices, it does not matter if we find the remaining edges. In other words, some false negative instances do not hurt, but we do want to find the edges that keep the graph connected. On the other hand, false positive instances are always not desired because it will add wrong connections.

Suppose there are m_k and m_r coreferential links in the key and the response, respectively, and a coreference resolution system successfully predicts n correct links. The recall and the precision are then $\frac{n}{m_k}$ and $\frac{n}{m_r}$, respectively.

The learnt classifier predicts false positive and false negative instances during testing. For a false positive instance, if we could successfully predict it as negative, the recall is unchanged, but the precision will be $\frac{n}{m_r-1}$, which is higher than the original precision $\frac{n}{m_r}$. For a false negative instance, it is more subtle. If the two markables in the instance are determined to be in the same coreference chain by the clustering algorithm, it does not matter whether we predict this instance as positive or negative, i.e., this false negative does not change the F-measure of the evaluation metric at all. If the two markables are not in the same coreference chain under the clustering, in case that we can predict it as positive, the recall will be $\frac{n+1}{m_k}$, which is higher than the original recall $\frac{n}{m_k}$, and the precision will be $\frac{n+1}{m_r+1}$, which is higher than the original precision $\frac{n}{m_r}$, as $n < m_r$.

In both cases, the F-measure improves. If we can instruct the learning algorithm to pay more attention to these false positive and false negative instances and to predict them correctly by assigning them more weight[1], we should be able to improve the F-measure.

[1] Assigning higher weights to the wrongly classified instances and assigning lower weights to the correctly classified instances have the same effect. Hence, in this thesis, we assign higher weights to the wrongly classified instances.

3.3.2 Beam Search

To instruct the learning algorithm to pay more attention to the wrongly classified instances, initially all instances are equally weighted, and we then iteratively assign more weights to the hard-to-classify pairs using a beam search algorithm.

The goal is to search for a set of weights to assign to the training instances such that the classifier trained on the weighted training instances gives the maximum coreference metric score when evaluated on the training instances. Each search state corresponds to a set of weighted training instances, a classifier trained on the weighted training instances minimizing misclassifications, and the F-measure of the classifier when evaluated on the weighted training instances using the chosen coreference evaluation metric.

Typically there are a large number of wrongly classified instances. To each wrongly classified instance, we could rectify or not rectify it. We can choose any subsets of wrongly classified instances to rectify. The number of combinations of choosing the instances to rectify is exponential, not to mention that we have many different ways to rectify each instance. It is impractical to explore all possibilities. We simplify the search by using a binary search tree. In the search tree, each search state is a node. The root of the search tree is the initial search state where all the training instances have identical weights of one. Each search state s can expand into two different children search states s_l (left child) and s_r (right child). s_l and s_r correspond to assigning higher weights to the false positive and false negative training instances in s, respectively. An expanded node always has two children in the binary search tree.

The binary search tree can grow infinitely. We use beam search to narrow down the search space. During the search, all nodes are sorted in descending order of F-measure. Only the top M nodes are kept, and the remaining nodes are discarded. The discarded nodes can either be leaf nodes or non-leaf nodes. The iterative algorithm stops when all the

nodes in the beam are non-leaf nodes, i.e., all the nodes in the beam have been expanded, and expanding the nodes in the beam will not improve the evaluation metric score.

Figure 3.1 shows an example of a binary search tree. Initially, the tree has only one node: the root (node 1 in the figure). In each iteration, the algorithm expands all the leaf nodes in the beam. For example, in the first iteration, node 1 is expanded to generate node 2 and 3, which correspond to adding weights to false positive and false negative training instances, respectively. Node 5 is discarded because of low F-measure, and it will not be expanded to generate children in the binary search tree.

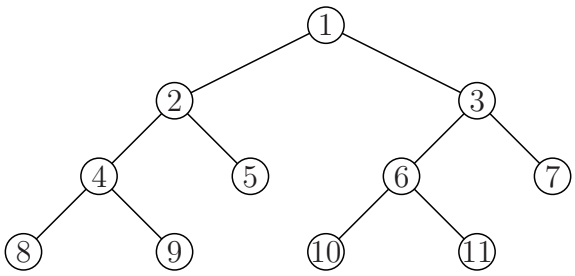

Figure 3.1: An example of a binary search tree

3.3.3 The Algorithm

We have so far described instance weighting and beam search to maximize the evaluation metric score. Combining them, we come up with the maximum metric score training algorithm. An overview of the algorithm is shown in Algorithm 3.3. The formal and detailed description of the algorithm is shown in Algorithm 3.4. In the algorithm, we assume that there are N texts T_1, T_2, ..., T_N in the training data set. m_{ki} and m_{kj} are the ith and the jth markable in the text T_k, respectively. Hence, for all $i < j$, $(m_{ki}, m_{kj}, w_{kij})$ is a training instance for the markable pair (m_{ki}, m_{kj}), in which w_{kij} is the weight of the instance. Let

Algorithm 3.3 Overview of the maximum metric score training (MMST) algorithm

```
initialization
repeat
  for each unexpanded beam node do
    classify (predict)
    update false positive instances
    update false negative instances
  end for
  beam pruning
until all nodes in the beam have been expanded
return the classifier
```

L_{kij} and L'_{kij} be the true and predicted label of the pair (m_{ki}, m_{kj}), respectively. Let W, C, F, and E be the set of weights $\{w_{kij}|1 \leq k \leq N, i < j\}$, the classifier, the F-measure, and a Boolean indicator of whether the search state has been expanded, respectively. Finally, M is the beam size, and δ controls how much we update the weights in each iteration.

In the algorithm, we train a classifier by weighting all instances equally in line (4) to (8). In the loop, we do coreference resolution using the current classifier in line (14) to (15), update the weights of false positive instances in line (16) to (24), and update the weights of false negative instances in line (25) to (34). This generates two classifiers to be evaluated in the next iteration. According to our beam search setting, only the top M classifiers are kept in each iteration. The loop continues until the top M classifiers do not change. The best classifier is then returned as the output of the algorithm.

3.4 Experiments

In this section, we describe the experimental settings and report the results of both the baseline systems and the results by the proposed MMST approach.

Algorithm 3.4 The maximum metric score training (MMST) algorithm

INPUT: $T_1, T_2, \ldots, T_N$
OUTPUT: classifier C

$w_{kij} \leftarrow 1$, for all $1 \leq k \leq N$ and $i < j$
$C \leftarrow$ train($\{(m_{ki}, m_{kj}, w_{kij}) | 1 \leq k \leq N, i < j\}$)
$F \leftarrow$ resolve and evaluate $T_1, \ldots, T_N$ with C
$E \leftarrow false$
BEAM$\leftarrow \{(W, C, F, E)\}$
repeat
  BEAM$' \leftarrow \{\}$
  for all (W, C, F, E) in BEAM **do**
    BEAM$' \leftarrow$ BEAM$' \bigcup \{(W, C, F, true)\}$
    if E=false **then**
      predict all L'_{kij} with C ($1 \leq k \leq N, i < j$)
      cluster into coreference chains based on L'_{kij}
      $W' \leftarrow W$
      for all $1 \leq k \leq N, i < j$ **do**
        if $L_{kij} = false$ and $L'_{kij} = true$ **then**
          $w'_{kij} \leftarrow w'_{kij} + \delta$
        end if
      end for
      $C' \leftarrow$ train($\{(m_{ki}, m_{kj}, w'_{kij}) | 1 \leq k \leq N, i < j\}$)
      $F' \leftarrow$ resolve and evaluate $T_1, \ldots, T_N$ with C'
      BEAM$' \leftarrow$ BEAM$' \bigcup \{(W', C', F', false)\}$
      $W'' \leftarrow W$
      for all $1 \leq k \leq N, i < j$ **do**
        if $L_{kij} = true$ and $L'_{kij} = false$ and
          $Chain(m_{ki}) \neq Chain(m_{kj})$ **then**
            $w''_{kij} \leftarrow w''_{kij} + \delta$
        end if
      end for
      $C'' \leftarrow$ train($\{(m_{ki}, m_{kj}, w''_{kij}) | 1 \leq k \leq N, i < j\}$)
      $F'' \leftarrow$ resolve and evaluate $T_1, \ldots, T_N$ with C''
      BEAM$' \leftarrow$ BEAM$' \bigcup \{(W'', C'', F'', false)\}$
    end if
  end for
  BEAM$\leftarrow$ BEAM$'$
  sort BEAM in descending order of F, keep top M elements
until for all E of all elements in BEAM, $E = true$
return C, from the top element (W, C, F, E) of BEAM

3.4.1 Experimental Setup

The Coreference Resolution System

In the literature, some previous work on coreference resolution assumed that gold standard markables were known and resolved coreferential relations only amongst the gold standard markables. In this thesis, we did not assume gold standard markables and worked directly on raw text input.

Stoyanov *et al.* (2009) presented the Reconcile package[2], a publicly available coreference resolution toolkit, that accepts raw text input and reported state-of-the-art coreference resolution results. Reconcile employs a comprehensive set of 62 features to represent each training instance, including lexical, proximity, grammatical, and semantic features. To ensure that the improvement of our proposed approach is not because of a lower baseline, we used the Reconcile package in our experiments, and implemented the proposed MMST algorithm on top of it.

In Reconcile, the raw texts are processed with a sequence of preprocessing components, including sentence segmentation, tokenization, part-of-speech tagging, syntactic parsing, and named entity recognition. In our experiments, sentence segmentation, tokenization, and part-of-speech tagging were performed using the OpenNLP toolkit[3]; syntactic parsing was performed using the Berkeley Parser[4]; and named entity recognition was performed using the Stanford NER[5,6]. Markable extraction was as defined in the task definition of each individual corpus.

[2] `http://www.cs.utah.edu/nlp/reconcile/`
[3] `http://opennlp.sourceforge.net/`
[4] `http://code.google.com/p/berkeleyparser/`
[5] `http://nlp.stanford.edu/ner/`
[6] Besides the pre-trained NER model that came with Stanford NER, Stoyanov *et al.* (2009) also used the NER model from LLNL (Lawrence Livermore National Laboratory). However, since we do not have access to the LLNL NER model, in this thesis, we used only the pre-trained NER model that came with the Stanford NER in the experiments.

Features for Coreference Resolution

In our experiments, we used two different feature sets for coreference resolution. The first feature set in our experiments consists of the same 62 features in Stoyanov *et al.* (2009). In the rest of this chapter, this feature set is referred to as "RFS" (standing for "Reconcile Feature Set").

In Zhao and Ng (2010), we used the BART package[7], another open source coreference resolution toolkit presented by Versley *et al.* (2008a), in the experiments. BART uses an extended feature set and tree kernel support vector machines (SVM-light-TK, (Moschitti, 2006)) under the Soon *et al.* (2001) training and testing framework. In Zhao and Ng (2010), the features we used were identical to the features output by the preprocessing code of BART reported in Versley *et al.* (2008a), except that we did not use their tree-valued and string-valued features. To have a further comparison, in this thesis we also used the feature set in Zhao and Ng (2010) but with the Reconcile package. In the rest of this chapter, this feature set is referred to as "BFS" (standing for "BART Feature Set").

We describe the features of the two feature sets as follows. In both feature sets, we define m_i and m_j as the first and second markable under consideration, respectively.

RFS

1. Agreement: T if m_i and m_j agree on both number and gender; F if they disagree on either number or gender; NA if gender or number is unknown for m_i or m_j.
2. Alias: T if m_i is an alias of m_j; F otherwise.
3. AlwaysCompatible: Always T.
4. Animacy: T if m_i and m_j agree on animacy; F otherwise.

[7] `http://www.sfs.uni-tuebingen.de/~versley/BART/`

5. Appositive: T if m_i and m_j are in an appositive construction; F otherwise.

6. Binding: T if m_i and m_j do not violate conditions B and C in Chomsky's binding theory (Chomsky, 1981); F otherwise.

7. BothEmbedded: T if both m_i and m_j are embedded; NA if only one is; F if neither is.

8. BothInQuotes: T if both m_i and m_j are inside of quotes; NA if only one is; F if neither is.

9. BothPronouns: T if both m_i and m_j are pronouns; NA if only one is; F if neither is.

10. BothProperNouns: T if both m_i and m_j are proper nouns; NA if only one is; F if neither is.

11. BothSubjects: T if both m_i and m_j are in the subject position relative to a verb clause; NA if only one is; F if neither is.

12. ClosestComp: T if m_i and m_j are compatible and the closest; F otherwise.

13. ConsecutiveSentences: T if m_i and m_j are in consecutive sentences; F otherwise.

14. Constraints: T if m_i and m_j are compatible in Gender, Number, Contraindices, Animacy, Pronoun, and ContainsPN; F otherwise.

15. ContainsPN: F if both m_i and m_j contain proper names and contain no words in common; T otherwise.

16. Contraindices: F if m_i and m_j violate either 1) two NPs separated by a preposition cannot be coindexed; or 2) two non-pronominal NPs separated by a non-copular verb cannot be coindexed; T otherwise.

17. Definite1: T if m_i starts with "the"; F otherwise.

18. Definite2: T if m_j starts with "the"; F otherwise.

19. Demostrative2: T if m_j starts with a demonstrative; F otherwise.

20. Embedded1: T if m_i is an embedded or nested NP; F otherwise.

21. Embedded2: T if m_j is an embedded or nested NP; F otherwise.

22. Gender: T if m_i and m_j agree in gender; F if they disagree; NA if the gender information cannot be determined.

23. HeadMatch: T if the head nouns of m_i and m_j match; F otherwise.

24. IAntes: T if one of m_i and m_j is the pronoun "I" and the other one is determined to be the quoted speaker of the text containing the "I" pronoun by a rule; F otherwise.

25. Indefinite: F if m_j is an indefinite and is not an appositive; T otherwise.

26. Indefinite1: F if m_j is an indefinite and is not an appositive; T otherwise. Similar to Indefinite, but use a slightly different way to determine if markable is indefinite.

27. InQuote1: T if m_i is inside of a quote; F otherwise.

28. InQuote2: T if m_j is inside of a quote; F otherwise.

29. MaximalNP: F if m_i and m_j have the same maximal NP projection; T otherwise.

30. Modifier: T if the prenominal modifiers of one of m_i and m_j are a subset of the prenominal modifiers of the other; F otherwise.

31. Number: T if m_i and m_j agree in number; F if they disagree; NA if the number information cannot be determined.

32. PairType: The value of this feature is t_i–t_j, where t_i and t_j are the types of m_i and m_j, respectively. The values of t are: n if m is a proper name; p if it is a pronoun; d if it is definite; and i if it is indefinite.

33. ParNum: The distance between m_i and m_j in terms of paragraphs.

34. PNStr: T if both m_i and m_j are proper names and the same string; F otherwise.

35. PNSubStr: T if both m_i and m_j are proper names and one is a substring of the other; F otherwise.

36. Prednom: T if m_i and m_j form a predicate nominal construction; F otherwise.

37. ProComp: T if m_i and m_j are both pronouns and are compatible in gender, number, and person; F otherwise.

38. Pronoun1: T if m_i is a pronoun; F otherwise.

39. Pronoun2: T if m_j is a pronoun; F otherwise.

40. Pronoun: F if m_i is a pronoun and m_j is not; T otherwise.

41. ProperName: F if m_i and m_j are both proper names and share no words in common; T otherwise.

42. ProperNoun: F if m_i and m_j are both proper nouns and share no words in common; T otherwise.

43. ProResolve: T if one of m_i and m_j is a pronoun and the other is its antecedent according to a rule-based algorithm; F otherwise.

44. ProStr: T if m_i and m_j are both pronouns and their strings match exactly; F otherwise.

45. Quantity: T if m_i and m_j form a pattern "sum of money" (e.g. loss of one thousand); F otherwise.

46. RuleResolve: T if m_i and m_j are coreferential according to a rule-based algorithm; F otherwise.

47. SameParagraph: T if m_i and m_j are in the same paragraph; F otherwise.

48. SameSentence: T if m_i and m_j are in the same sentence; F otherwise.

49. SentNum: The distance between m_i and m_j in terms of sentences.

50. SoonStr: T if m_i and m_j match after discarding uninformative words; F otherwise.

51. Span: F if one of m_i and m_j spans the other; T otherwise.

52. SubClass: T if the WordNet (Fellbaum, 1998) class of one of m_i and m_j is a subclass of the WorNet class of the other; F otherwise.

53. Subject1: T if m_i is a subject; F otherwise.

54. Subject2: T if m_j is a subject; F otherwise.

55. Syntax: F if m_i and m_j have incompatible values for Binding, Contraindices, Span, or MaximalNP; T otherwise.

56. WNSynonyms: T if m_i and m_j are WordNet synonyms; F otherwise.

57. WordNetClass: T if m_i and m_j have the same WordNet class; F otherwise.

58. WordNetDist: The distance in the WordNet synset tree between m_i and m_j.

59. WordNetSense: The first WordNet sense that m_i and m_j share.

60. WordOverlap: T if the intersection of the content words of m_i and m_j is not empty; F otherwise.

61. WordStr: T if m_i and m_j are both non-pronominal and the strings match; F otherwise.

62. WordsSubStr: T if m_i and m_j are both non-pronominal and one of them is a proper substring of the other with respect to content words; F otherwise.

BFS

1. Alias: T if m_i is an alias of m_j; F otherwise.

2. AnaIsDefinite: T if m_j starts with "the"; F otherwise.

3. AnaIsDemNominal: T if m_j is a demonstrative nominal; F otherwise.

4. AnaIsDemostrative: T if m_j starts with a demonstrative; F otherwise.

5. AnaIsDemPronoun: T if m_j is a demonstrative pronoun; F otherwise.

6. AnaIsPN: T if m_j is a proper noun; F otherwise.

7. AnaIsPronoun: T if m_j is a pronoun; F otherwise.

8. AnaIsReflPronoun: T if m_j is a reflexive pronoun; F otherwise.

9. AnaIsPersPronoun: T if m_j is a personal pronoun; F otherwise.

10. AnaIsPossPronoun: T if m_j is a possessive pronoun; F otherwise.

11. AnaSemClass: The semantic class of m_j.

12. AntIsDefinite: T if m_i starts with "the"; F otherwise.

13. AntIsFirstMention: T if m_i is the first markable in the utterance; F otherwise.

14. AntIsPN: T if m_i is a proper noun; F otherwise.

15. AntIsPronoun: T if m_i is a pronoun; F otherwise.

16. AntSemClass: The semantic class of m_i.

17. Appositive: T if m_i and m_j are in an appositive construction; F otherwise.

18. AreProperNoun: T if both m_i and m_j are proper nouns; NA if only one is; F if neither is.

19. DistLast: T if m_i and m_j are in consecutive sentences; F otherwise.

20. DistSame: T if m_i and m_j are in the same sentence; F otherwise.

21. Gender: T if m_i and m_j agree in gender; F if they disagree; NA if the gender information cannot be determined.

22. IsWikiAlias: Similarity between m_i and m_j based on the alias information encoded in Wikipedia[8] hyperlinks.

23. IsWikiRedir: Similarity between m_i and m_j based on Wikipedia page redirection.

24. IsWikiLists: Similarity between m_i and m_j based on Wikipedia list pages.

25. Number: T if m_i and m_j agree in number; F if they disagree; NA if the number information cannot be determined.

26. SemanticCompatibility: T if m_i and m_j are compatible in semantic class; F if not compatible; NA if either semantic classes is unknown.

[8] `http://www.wikipedia.org/`

		MUC6	MUC7	BNEWS	NPAPER	NWIRE
# of Texts	Training	30	30	216	76	130
	Test	30	20	51	17	29
# of Words	Training	13,404	16,313	66,630	68,463	70,820
	Test	14,117	11,107	17,464	17,350	16,781
# of Markables	Training	4,445	4,917	20,467	21,492	22,011
	Test	4,546	3,236	5,428	5,153	5,230
Recall of Markables	Training	91.9	90.2	95.5	94.5	94.2
	Test	92.2	87.0	95.6	95.5	95.3
# of Instances	Training	505,534	418,450	1,763,817	3,365,680	3,008,303
	Test	614,444	282,256	436,347	871,314	681,092

Table 3.1: Statistics of the two MUC and the three ACE corpora.

27. StringMatch: T if m_i and m_j match after discarding uninformative words; F otherwise.

28. WebMatch: T if m_i and m_j match according to some patterns mined from the web.

Corpora

In the experiments, we used all five commonly used evaluation corpora for coreference resolution, namely the two MUC corpora (MUC6 and MUC7) and the three ACE corpora (BNEWS, NPAPER, and NWIRE). The MUC6 and the MUC7 corpora were defined in the DARPA Message Understanding Conference (MUC-6, 1995; MUC-7, 1998). The dry-run texts were used as the training data sets. In both corpora, each training data set contains 30 texts. The test data sets for MUC6 and MUC7 consist of the 30 and 20 formal evaluation texts, respectively. The ACE corpora were defined in NIST Automatic Content Extraction phase 2 (ACE-2) (NIST, 2002). The three data sets are from different news sources: broadcast news (BNEWS), newspaper (NPAPER), and newswire (NWIRE). Although the official test sets are only available to the ACE participants, the training sets and development test sets are available to the public. They were used as our training set and test set, respectively.

The BNEWS, NPAPER, and NWIRE data sets contain 216, 76, and 130 training texts, and 51, 17, and 29 test texts, respectively. The statistics of all five corpora are tabulated in Table 3.1. The numbers of training instances in Table 3.1 are the numbers of all possible pairs of markables in the data sets.

Evaluation Metrics

Since we used automatically extracted markables, it is possible that some extracted markables and the gold standard markables are unmatched, or *twinless* as defined in Stoyanov *et al.* (2009). How to use the B-CUBED metric to evaluate twinless markables has been explored recently. In this thesis, we adopted the B^3all variation proposed by Stoyanov *et al.* (2009), which retains all twinless markables.

Learning Algorithms

In Zhao and Ng (2010), we used the C4.5 decision tree learning algorithm (Quinlan, 1993) in the experiments. Decision tree learning uses a tree to support decision making. It is a widely used learning algorithm in research on coreference resolution and many other natural language processing problems. It starts from the root of the tree and recursively splits the data set until most of the training instances in the same node are of the same class. At each node of the tree, C4.5 chooses a feature that results in the highest information gain, which is the difference of information entropy before and after the split. Information entropy I is computed as follows:

$$I(P(c_1), P(c_2), \ldots, P(c_n)) = \sum_{i=1}^{n} -P(c_i) \log P(c_i) \tag{3.2}$$

where c_1, c_2, . . ., c_n are the classes of the training instances, and n is the number of different classes.

Originally, $P(c_i)$, the probability of class c_i, is the number of training instances that belong to class c_i over the number of all training instances:

$$P(c_i) = \frac{|c_i|}{\sum_{k=1}^{n} |c_k|} \tag{3.3}$$

where $|c_i|$ is the number of training instances that belong to class c_i.

With instance weighting, however, this is computed as the sum of weights of the instances that belong to c_i, over the sum of weights of all training instances:

$$P(c_i) = \frac{\sum_{k \in c_i} w_k}{\sum_{k=1}^{N} w_k} \tag{3.4}$$

where w_k is the weight of the k-th instance, and N is the total number of instances.

It is easy to see that when all weights w_k are 1, Equation 3.4 is the same as Equation 3.3. Furthermore, the effect of setting the weight of a training instance to n is equivalent to duplicating the instance $n - 1$ times, i.e., a total of n instances.

Besides decision tree, maximum entropy (ME) modeling (Berger *et al.*, 1996) is another widely used learning algorithm in the literature on coreference resolution. Also known as logistic regression in the community of machine learning (Hastie *et al.*, 2001; Wasserman, 2004), a maximum entropy model predicts the class label of a test instance by maximizing the conditional probability $p(c|x)$.

In maximum entropy learning, the goal is to learn a function f that minimizes the expected loss with respect to the distribution of the training data $P(x, y)$:

$$f^* = \arg\min_{f \in H} \frac{1}{N} \sum_{i=1}^{N} L(x_i, y_i, f) \tag{3.5}$$

where $L(x, y, f)$ is the loss function, x and y are the features and label of an instance, respectively, H is the hypothesis space, and N is the total number of instances.

Similar to instance weighting in decision tree learning, with instance weighting, Equation 3.5 will become

$$f^* = \arg\min_{f \in H} \frac{1}{\sum_{k=1}^{N} w_k} \sum_{i=1}^{N} w_i L(x_i, y_i, f) \tag{3.6}$$

It is also easy to see that when all weights w_k are 1, Equation 3.6 is the same as Equation 3.5. Similarly, the effect of setting the weight of a training instance to n is equivalent to duplicating the instance $n-1$ times, i.e., a total of n instances.

In the experiments for maximum metric score training, we used both decision tree models and maximum entropy models. For decision tree learning, we used the WEKA implementation of the C4.5 decision tree (Witten and Frank, 2005), while for maximum entropy learning, we used the maximum entropy modeling implemented in the DALR package (Jiang and Zhai, 2007). However, our proposed approach is able to be applied to any supervised machine learning algorithms.

Putting Things Together

Thus far, we have described two feature sets, five data sets, two evaluation metrics, and two learning algorithms in the experimental settings. In total, there are 40 different combinations. In our experiments, we evaluated all five data sets on two evaluation metrics. As for the feature sets and learning algorithms, we tried the "BFS + decision tree" combination, as in Zhao and Ng (2010), and the "RFS + maximum entropy" combination, as we will use in Chapter 4. To bridge the two combinations, we tried the "BFS + maximum entropy" combination as well [9].

Specifically, results reported in Table 3.2 to 3.5 were obtained using BFS and decision

[9] We do not report "RFS + decision tree" because it requires very large memory in the experiments but we do not have a proper machine to do so.

tree learning; results reported in Table 3.6 to 3.9 were obtained using BFS and maximum entropy learning; and results reported in Table 3.10 to 3.13 were obtained using RFS and maximum entropy learning.

3.4.2 The Baseline Systems

We included state-of-the-art coreference resolution systems in the literature for comparison.

Since we used the Reconcile package in our experiments, we included the results of the original Reconcile system reported in Stoyanov *et al.* (2009) as the first system for comparison.

We used the BART package in Zhao and Ng (2010). In this thesis, we also used the feature set derived from BART. Thus, we also included the results of the original BART system (with its extended feature set and SVM-light-TK, as reported in Versley *et al.* (2008a)) as another system for comparison.

Stoyanov *et al.* (2009) reported only the results on the MUC6, MUC7, and NWIRE data sets, but not the BNEWS and NPAPER data sets. Versley *et al.* (2008a) reported only the results on the three ACE data sets with the MUC evaluation metric. Since we used all the five data sets in our experiments, for fair comparison, we also included the MUC results reported in Ng (2004a). To the best of our knowledge, Ng (2004a) was the only prior work which reported MUC metric scores on all the five data sets.

The MUC metric scores of Stoyanov *et al.* (2009), Versley *et al.* (2008a), and Ng (2004a) are listed in the rows "Stoyanov *et al.* 09", "Versley *et al.* 08" and "Ng 04", respectively, in Table 3.2, 3.3, 3.6, 3.7, 3.10, and 3.11.

For the B-CUBED metric, we included Stoyanov *et al.* (2009) and Ng (2005)[10] for comparison. The scores are listed in the rows "Stoyanov *et al.* 09" and "Ng 05", respectively,

[10]How Ng (2005) interpreted the B-CUBED metric for raw text inputs is not stated in his paper. However, for comparison, we still list his results here.

in Table 3.4, 3.5, 3.8, 3.9, 3.12, and 3.13.

Besides the state-of-the-art results reported in the literature, we built our own baselines to see how the proposed maximum metric score training algorithm improves the performance of coreference resolution. Our baselines that follow the Soon *et al.* (2001) framework are shown in the rows "*SNL*-Style Baseline" in Table 3.2 to 3.13. Our proposed MMST algorithm trains and tests on all pairs of markables. To show the effectiveness of weight updating of MMST, we also built baselines which train and test on all pairs. The performances of these systems are shown in the rows "*All*-Style Baseline" in Table 3.2 to 3.13.

The results show that our baseline systems are comparable to the state of the art in the literature. When compared against Stoyanov *et al.* (2009), which trained and tested on all pairs and used the averaged perceptron algorithm (Freund and Schapire, 1999) as their learning algorithm, our *All*-style baseline achieved F-measures comparable to theirs, although we do not have access to their LLNL NER model. In some combinations, our *All*-style baseline outperformed Stoyanov *et al.* (2009). For example, in Table 3.4, the *All*-style baseline has an F-measure of 65.2% on MUC7, which is 2.3% higher than the 62.9% F-measure in Stoyanov *et al.* (2009). To achieve good performance, Versley *et al.* (2008a) used the time-consuming tree kernel support vector machines (SVM-light-TK), while Ng (2004a) and Ng (2005) used additional pipelined components in their system, e.g., anaphoricity determination. Although our baseline systems did not use additional components or techniques, in some combinations, we achieved higher performance. For example. in Table 3.11, the *SNL*-style baseline and the *All*-style baseline achieved F-measures of 0.9% and 4.9% higher than Ng (2004a) on NWIRE, respectively.

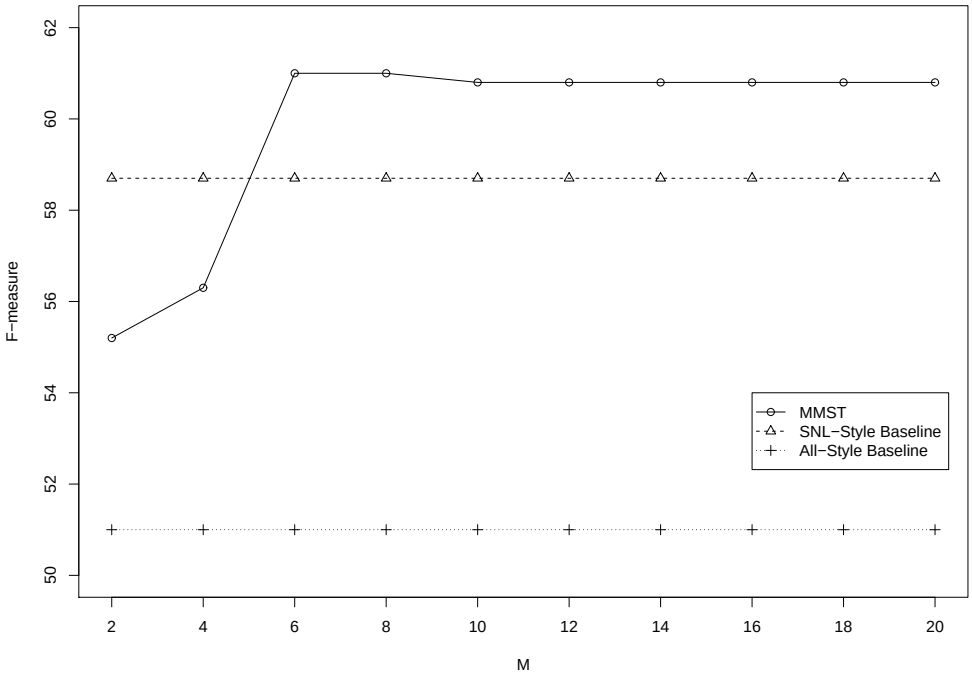

Figure 3.2: Tuning M on the held-out development set

3.4.3 Results Using Maximum Metric Score Training

Next, we show the results of using the proposed maximum metric score training algorithm. From the description of the algorithm, it can be seen that there are two parameters in the algorithm. One parameter is M, the size of the beam. The other parameter is δ, which controls how much we increase the weight of a training instance in each iteration.

Since the best M and δ for the MUC evaluation metric were not known, we used held-out development sets to tune the parameters. Specifically, we trained classifiers with different combinations of M and δ on a development training set, and evaluated their performances on a development test set. In our experiments, the development training set contained 2/3 of the texts in the training set of each individual corpus, while the development test set contained the remaining 1/3 of the texts. After picking the best M and δ

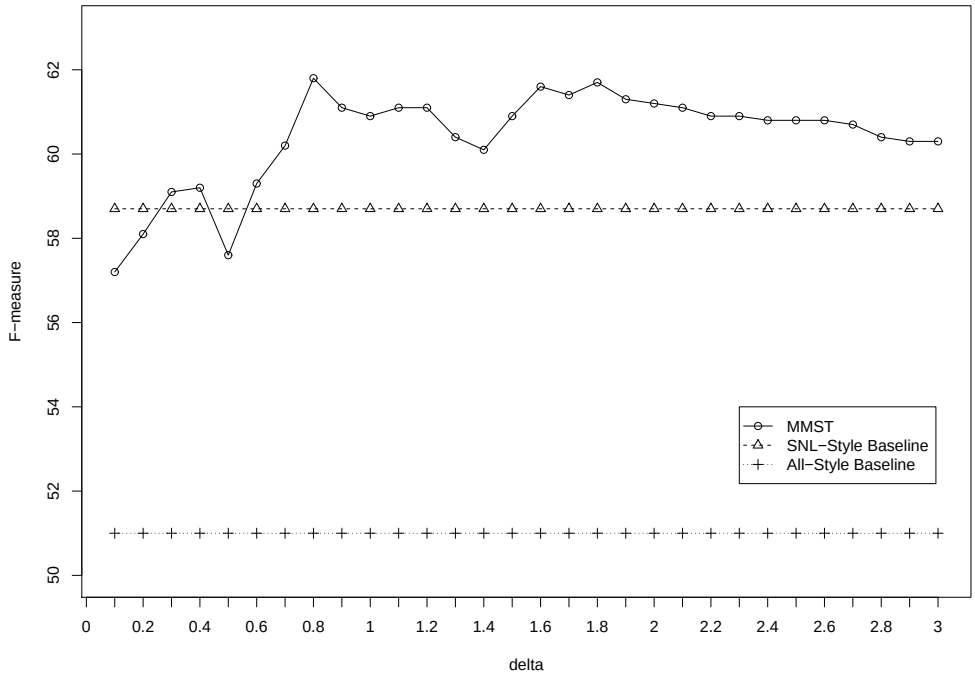

Figure 3.3: Tuning δ on the held-out development set

values, we trained a classifier on the entire training set with the chosen parameters. The learnt classifier was then applied to the test set.

To limit the search space, we tuned the two parameters sequentially. First, we fixed $\delta = 1$, which is equivalent to duplicating each training instance once in decision tree and maximum entropy learning, and evaluated $M = 2,\ 4,\ 6,\ \ldots,\ 20$. After choosing the best M that corresponded to the maximum F-measure, we fixed the value of M, and evaluated $\delta = 0.1,\ 0.2,\ 0.3,\ \ldots,\ 3.0$. Take MUC6, with the MUC evaluation metric, using BFS and decision tree learning, as an example. The results of tuning M under these settings, are shown in Figure 3.2. The maximum F-measure was obtained when $M = 6$ or $M = 8$. On all the different M values we have tried, MMST outperforms the *All*-style baseline on the development test set. When $M > 4$, MMST outperforms the *SNL*-style baseline on

the development test set. We then fixed $M = 6$, and evaluated different δ values. The results are shown in Figure 3.3. The best F-measure was obtained when $\delta = 0.8$. Again, on all the different δ values we have tried, MMST outperforms the *All*-style baseline on the development test set. It also outperforms the *SNL*-style baseline with most δ values.

The rows "MMST" in Table 3.2 to 3.13 show the performance of MMST on the test sets, with the tuned parameters indicated. In our experiments, the statistical significance test was conducted as in Chinchor (1995). * and ** stand for $p < 0.05$ and $p < 0.01$ over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ over the *All*-style baseline, respectively.

In all the different combinations of settings, MMST outperforms both the *SNL*-style baseline and the *All*-style baseline, with the only exception on NPAPER with B-CUBED evaluation metric, using RFS and maximum entropy learning, MMST outperforms the *SNL*-style baseline, but is 0.2% lower than the *All*-style baseline in F-measure. For example, with MUC evaluation metric, using BFS and decision tree learning, when compared to the *All*-style baseline, MMST gained 4.8%, 0.5%, 8.5%, 3.2%, and 2.1% improvement in F-measure on MUC6, MUC7, BNEWS, NPAPER, and NWIRE, respectively. Furthermore, the experimental results clearly show that MMST achieves not only consistent, but also statistically significant improvements over at least one of the baselines, and in most cases both baselines (the *SNL*-style baseline and the *All*-style baseline) in the different settings.

3.4.4 Analysis

To see how MMST actually updates the weight, we use the MUC metric, BFS and decision tree learning, as an example. Under the experimental settings, it took 5 – 7 iterations for MMST to stop on the five data sets. The numbers of explored states in the binary search tree, including the root, were 31, 19, 9, 13, and 13 on MUC6, MUC7, BNEWS, NPAPER,

Model	MUC6 R	MUC6 P	MUC6 F	MUC7 R	MUC7 P	MUC7 F
Stoyanov *et al.* 09			**70.4**			58.2
Ng 04	75.8	61.4	67.9	64.2	60.2	**62.1**
SNL-Style Baseline	68.9	49.4	57.5	56.4	55.8	56.1
All-Style Baseline	54.6	71.3	61.8	49.6	78.9	60.9
MMST	69.5	63.9	66.6**††	57.7	65.8	61.4**
	$M = 6,\ \delta = 0.8$			$M = 6,\ \delta = 1.7$		

Table 3.2: Results for the two MUC corpora with MUC evaluation metric, using BFS and decision tree learning. Boldface indicates the best results. * and ** stand for $p < 0.05$ and $p < 0.01$ significance levels over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ significance levels over the *All*-style baseline, respectively.

Model	BNEWS R	BNEWS P	BNEWS F	NPAPER R	NPAPER P	NPAPER F	NWIRE R	NWIRE P	NWIRE F
Stoyanov *et al.* 09									65.8
Versley *et al.* 08	60.7	65.4	63.0	64.1	67.7	65.8	60.4	65.2	62.7
Ng 04	63.1	67.8	65.4	73.5	63.3	68.0	53.1	60.6	56.6
SNL-Style Baseline	61.0	42.7	50.2	69.3	47.0	56.0	66.1	50.4	57.2
All-Style Baseline	49.3	77.6	60.3	56.9	78.6	66.0	60.5	74.9	66.9
MMST	73.2	64.9	**68.8****††	67.5	71.1	**69.2****††	68.1	69.9	**69.0****†
	$M = 2,\ \delta = 0.9$			$M = 4,\ \delta = 1.0$			$M = 2,\ \delta = 0.3$		

Table 3.3: Results for the three ACE corpora with MUC evaluation metric, using BFS and decision tree learning. Boldface indicates the best results. * and ** stand for $p < 0.05$ and $p < 0.01$ significance levels over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ significance levels over the *All*-style baseline, respectively.

Model	MUC6			MUC7		
	R	P	F	R	P	F
Stoyanov *et al.* 09			**69.9**			62.9
SNL-Style Baseline	60.8	69.5	64.9	53.9	78.7	64.0
All-Style Baseline	54.1	84.5	66.0	50.3	92.5	65.2
MMST	62.7	75.8	68.6**	55.4	82.0	**66.1***
	$M = 8,\ \delta = 1.1$			$M = 2,\ \delta = 0.5$		

Table 3.4: Results for the two MUC corpora with B-CUBED evaluation metric, using BFS and decision tree learning. Boldface indicates the best results. * and ** stand for $p < 0.05$ and $p < 0.01$ significance levels over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ significance levels over the *All*-style baseline, respectively.

Model	BNEWS			NPAPER			NWIRE		
	R	P	F	R	P	F	R	P	F
Stoyanov *et al.* 09									77.3
Ng 05	57.0	77.1	65.6	62.8	71.2	66.7	59.3	75.4	66.4
SNL-Style Baseline	64.1	74.3	68.8	66.2	71.6	68.8	69.4	78.0	73.4
All-Style Baseline	59.1	94.5	72.7	62.4	91.2	74.1	68.9	90.1	78.1
MMST	72.3	83.4	**77.4****††	70.8	82.0	**76.0****††	73.7	84.8	**78.8****†
	$M = 6,\ \delta = 0.2$			$M = 6,\ \delta = 0.7$			$M = 6,\ \delta = 1.2$		

Table 3.5: Results for the three ACE corpora with B-CUBED evaluation metric, using BFS and decision tree learning. Boldface indicates the best results. * and ** stand for $p < 0.05$ and $p < 0.01$ significance levels over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ significance levels over the *All*-style baseline, respectively.

Model	MUC6 R	MUC6 P	MUC6 F	MUC7 R	MUC7 P	MUC7 F
Stoyanov *et al.* 09			**70.4**			58.2
Ng 04	75.8	61.4	67.9	64.2	60.2	**62.1**
SNL-Style Baseline	66.1	54.2	59.5	56.1	58.5	57.3
All-Style Baseline	43.1	77.5	55.4	42.9	80.2	55.9
MMST	75.5	52.9	62.2*††	59.7	61.0	60.3*††
	$M = 6,\ \delta = 1.6$			$M = 10,\ \delta = 2.2$		

Table 3.6: Results for the two MUC corpora with MUC evaluation metric, using BFS and maximum entropy learning. Boldface indicates the best results. * and ** stand for $p < 0.05$ and $p < 0.01$ significance levels over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ significance levels over the *All*-style baseline, respectively.

Model	BNEWS R	BNEWS P	BNEWS F	NPAPER R	NPAPER P	NPAPER F	NWIRE R	NWIRE P	NWIRE F
Stoyanov *et al.* 09									**65.8**
Versley *et al.* 08	60.7	65.4	63.0	64.1	67.7	65.8	60.4	65.2	62.7
Ng 04	63.1	67.8	65.4	73.5	63.3	68.0	53.1	60.6	56.6
SNL-Style Baseline	54.1	48.4	51.1	64.0	43.1	51.5	61.1	47.7	53.6
All-Style Baseline	38.6	74.5	50.9	52.5	71.7	60.6	44.7	77.6	56.7
MMST	74.7	58.9	**65.9****††	80.3	62.2	**70.1****††	69.0	61.6	65.1**††
	$M = 20,\ \delta = 1.6$			$M = 4,\ \delta = 1.7$			$M = 6,\ \delta = 2.3$		

Table 3.7: Results for the three ACE corpora with MUC evaluation metric, using BFS and maximum entropy learning. Boldface indicates the best results. * and ** stand for $p < 0.05$ and $p < 0.01$ significance levels over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ significance levels over the *All*-style baseline, respectively.

Model	MUC6			MUC7		
	R	P	F	R	P	F
Stoyanov *et al.* 09			**69.9**			62.9
SNL-Style Baseline	58.1	76.1	65.9	52.1	80.8	63.4
All-Style Baseline	43.7	92.6	59.4	45.0	94.2	60.9
MMST	69.0	67.5	68.2††	56.3	77.2	**65.1**††
	$M = 6,\ \delta = 1.0$			$M = 6,\ \delta = 1.5$		

Table 3.8: Results for the two MUC corpora with B-CUBED evaluation metric, using BFS and maximum entropy learning. Boldface indicates the best results. * and ** stand for $p < 0.05$ and $p < 0.01$ significance levels over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ significance levels over the *All*-style baseline, respectively.

Model	BNEWS			NPAPER			NWIRE		
	R	P	F	R	P	F	R	P	F
Stoyanov *et al.* 09									77.3
Ng 05	57.0	77.1	65.6	62.8	71.2	66.7	59.3	75.4	66.4
SNL-Style Baseline	59.5	81.0	68.6	59.5	70.3	64.5	65.0	78.6	71.1
All-Style Baseline	54.3	95.0	69.1	59.7	88.4	71.3	58.1	95.0	72.1
MMST	72.1	80.3	**76.0****††	72.9	76.1	**74.5****††	70.9	85.8	**77.6****††
	$M = 6,\ \delta = 2.6$			$M = 6,\ \delta = 1.0$			$M = 6,\ \delta = 1.9$		

Table 3.9: Results for the three ACE corpora with B-CUBED evaluation metric, using BFS and maximum entropy learning. Boldface indicates the best results. * and ** stand for $p < 0.05$ and $p < 0.01$ significance levels over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ significance levels over the *All*-style baseline, respectively.

Model	MUC6			MUC7		
	R	P	F	R	P	F
Stoyanov *et al.* 09			**70.4**			58.2
Ng 04	75.8	61.4	67.9	64.2	60.2	62.1
SNL-Style Baseline	70.4	57.9	63.5	59.2	59.9	59.6
All-Style Baseline	54.9	78.2	64.5	47.6	75.7	58.4
MMST	67.3	66.1	66.7**†	60.1	66.8	**63.3****††
	$M = 2,\ \delta = 1.7$			$M = 2,\ \delta = 1.5$		

Table 3.10: Results for the two MUC corpora with MUC evaluation metric, using RFS and maximum entropy learning. Boldface indicates the best results. * and ** stand for $p < 0.05$ and $p < 0.01$ significance levels over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ significance levels over the *All*-style baseline, respectively.

Model	BNEWS			NPAPER			NWIRE		
	R	P	F	R	P	F	R	P	F
Stoyanov *et al.* 09									**65.8**
Versley *et al.* 08	60.7	65.4	63.0	64.1	67.7	65.8	60.4	65.2	62.7
Ng 04	63.1	67.8	**65.4**	73.5	63.3	**68.0**	53.1	60.6	56.6
SNL-Style Baseline	64.7	49.3	56.0	67.6	50.5	57.8	66.3	50.9	57.5
All-Style Baseline	50.7	73.3	60.0	59.0	70.6	64.3	50.1	79.5	61.5
MMST	61.7	66.6	64.0**††	74.7	59.2	66.1**	65.7	61.9	63.7**
	$M = 8,\ \delta = 1.1$			$M = 6,\ \delta = 2.0$			$M = 10,\ \delta = 1.3$		

Table 3.11: Results for the three ACE corpora with MUC evaluation metric, using RFS and maximum entropy learning. Boldface indicates the best results. * and ** stand for $p < 0.05$ and $p < 0.01$ significance levels over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ significance levels over the *All*-style baseline, respectively.

Model	MUC6 R	MUC6 P	MUC6 F	MUC7 R	MUC7 P	MUC7 F
Stoyanov *et al.* 09			69.9			62.9
SNL-Style Baseline	64.9	77.3	70.5	55.5	80.4	65.7
All-Style Baseline	54.3	90.7	67.9	49.0	91.3	63.8
MMST	67.2	75.9	**71.3**††	61.7	77.2	**68.6****††
	$M = 4,\ \delta = 1.5$			$M = 6,\ \delta = 0.8$		

Table 3.12: Results for the two MUC corpora with B-CUBED evaluation metric, using RFS and maximum entropy learning. Boldface indicates the best results. * and ** stand for $p < 0.05$ and $p < 0.01$ significance levels over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ significance levels over the *All*-style baseline, respectively.

Model	BNEWS R	BNEWS P	BNEWS F	NPAPER R	NPAPER P	NPAPER F	NWIRE R	NWIRE P	NWIRE F
Stoyanov *et al.* 09									77.3
Ng 05	57.0	77.1	65.6	62.8	71.2	66.7	59.3	75.4	66.4
SNL-Style Baseline	65.1	77.6	70.8	67.5	76.5	71.7	71.3	78.7	74.8
All-Style Baseline	60.1	92.4	72.8	65.6	87.7	**75.1**	63.9	94.2	76.2
MMST	64.6	88.2	**74.6****††	71.7	78.4	74.9**	69.5	88.3	**77.8****††
	$M = 6,\ \delta = 1.1$			$M = 6,\ \delta = 1.1$			$M = 6,\ \delta = 1.0$		

Table 3.13: Results for the three ACE corpora with B-CUBED evaluation metric, using RFS and maximum entropy learning. Boldface indicates the best results. * and ** stand for $p < 0.05$ and $p < 0.01$ significance levels over the *SNL*-style baseline, respectively. † and †† stand for $p < 0.05$ and $p < 0.01$ significance levels over the *All*-style baseline, respectively.

and NWIRE, respectively. It is instructive to find out the final weight of each instance. Take MUC6 under the above setting as an example, the numbers of positive instances with weight 1, 1.8, 2.6, 3.4, and 4.2 were 4,162, 1,286, 1,992, 549, and 1,592, respectively, while the numbers of negative instances with weight 1 and 1.8 were 495,068 and 885, respectively. Counting the weighted number of instances (e.g., an instance with weight 2 is equivalent to 2 instances), we have 20,209 positive and 496,661 negative training instances. This changes the ratio of the positive instances from 1.9% to 3.9%. As a by-product, MMST reduces data skewness, while using all possible NP pairs for training to keep as much information as possible.

MMST tries to focus on the "critical links" which, if rectified, will lead to a higher score of the coreference resolution evaluation metric. The change of weights of the "targeted" training instances is equivalent to the change of distribution of the training instances. This effectively changes the classification hypothesis to one that tends to yield higher evaluation metric score.

Here is a more intuitive example. In the sentence

> In a news release, ***the company*** said the new name more accurately reflects ***its*** focus on high-technology communications, including business and entertainment software, interactive media and wireless data and voice transmission.

the pronoun *its* is coreferential to the antecedent NP *the company* in gold standard annotation. The baseline classifier wrongly resolves *its* to another NP *the new name*, but with MMST, *its* is successfully resolved to *the company*. The baseline classifier made one false positive and one false negative prediction. Take the false positive instance as an example, it is resolved to the wrong antecedent mainly because *the new name* is the closest NP preceding *its* and is compatible to it. Because MMST has seen similar mistakes and learnt

to correct them during training, it is able to predict it correctly during testing, which the baseline classifier fails to do so.

In addition, our MMST approach improves upon state-of-the-art results on most of the five standard benchmark corpora (two MUC corpora and three ACE corpora), with both the link-based MUC metric and the mention-based B-CUBED metric, using BFS and RFS feature sets, and with decision tree learning or maximum entropy learning.

Last but not least, our approach performs all the F-measure maximization during training, and is very fast during testing, since the output of the MMST algorithm is a standard classifier. For example, on the MUC6 data set with the MUC evaluation metric, using BFS and decision tree learning, it took 51 minutes and 1 second for training and testing, respectively, on an Intel Xeon 2.4GHz machine.

3.5 Summary

In this chapter, we have presented MMST, a generic framework and novel approach comprising the use of instance weighting and beam search to train a classifier to maximize the chosen coreference metric score on the training corpus during training. Experimental results showed that MMST achieves statistically significant improvements over the *Soon*-style and *All*-style baselines on most of the five standard benchmark corpora (two MUC corpora and three ACE corpora), with both the link-based MUC metric and the mention-based B-CUBED metric, using BFS and RFS feature sets, and with decision tree learning or maximum entropy learning.

Chapter 4

Domain Adaptation for Coreference Resolution

In Chapter 1, we have pointed out that one of the most challenging obstacles in applying supervised learning approaches to coreference resolution is the difficulty in data annotation. It is much more time-consuming and expensive to annotate a corpus for coreference resolution than to annotate a corpus for other natural language processing tasks. Most existing annotated corpora for coreference resolution are in the newswire domain. To achieve good coreference resolution performance in a new domain, it is almost inevitable that we annotate some data. This raises the question of how to minimize the amount of data annotation needed while maintaining good coreference resolution performance. Although active learning has been successfully applied to other natural language processing and information retrieval tasks to reduce the amount of annotation needed, it remains an open problem how to apply active learning for coreference resolution, especially integrating it with domain adaptation.

In this chapter, we explore domain adaptation for coreference resolution from a source

domain that we have a large collection of annotated data, to a second target domain that we want good performance. We also integrate domain adaptation with active learning to reduce the effort of data annotation in coreference resolution while maintaining comparable performance. Our approach combines domain adaptation, active learning, and target domain instance weighting together to leverage the existing annotated corpora from newswire domain to reduce the cost of developing a coreference resolution system in biomedical domain. The approach achieves comparable coreference resolution performance on MEDLINE abstracts, but with a greatly reduced number of training instances that we need to annotate. To the best of our knowledge, our work is not only the first to use domain adaptation for coreference resolution, but also the first successful one to use active learning for coreference resolution.

The rest of this chapter is organized as follows. We first introduce the background of domain adaptation in coreference resolution in Section 4.1. Then we describe the domain adaptation and active learning techniques, as well as how to combine them together for coreference resolution in Section 4.2. Experimental results and analysis are presented in Section 4.3. Finally, we conclude the chapter in Section 4.4.

4.1 Background

4.1.1 Data Annotation in Coreference Resolution

In Chapter 1, we have explained that domain adaptation in coreference resolution traditionally needs much more effort than domain adaptation in other natural language processing and information retrieval tasks. In order to apply supervised machine learning approaches, one needs to collect a text corpus in the specific domain and annotate it as training data. Compared to many other NLP tasks, to annotate a corpus for coreference resolution, the

annotator needs to read and understand much more material, and hence it takes him or her much more time. Because of the nature of pairwise coreferential relations, the number of annotations in coreference resolution is $O(n^2)$, where n is the number of markables, compared to $O(n)$ number of annotations in many other NLP tasks. Cohen *et al.* (2010) reported that it took an average of 20 hours to annotate coreferential relations on a single document with an average length of 6,155 words, while an annotator could annotate 3,000 words per hour in POS tag annotation (Marcus *et al.*, 1993).

It is time-consuming and expensive to annotate new data sets for each new domain. If we want to save the efforts in applying coreference resolution from a resource-rich domain to a different domain that we want good performance, the direct way is to train a coreference resolution system on the resource-rich source domain and apply it to the target domain without any new data annotation. However, the domain differences make this direct application sub-optimal. In the next section, we explain this by introducing coreference resolution in the biomedical domain.

4.1.2 Coreference Resolution in the Biomedical Domain

A large body of prior research on coreference resolution focuses on texts in the newswire domain. Standardized data sets, such as the MUC and ACE data sets are widely used in the study of coreference resolution. We have used the corpora in newswire domain in our experiments in Chapter 3. There is a relatively small body of prior research on coreference resolution in non-newswire domains. Similar to other natural language processing problems, coreference resolution in different domains could differ significantly from one another. For example, coreference resolution in the newswire domain is quite different from coreference resolution in the biomedical domain.

Here is an example of coreference resolution in the biomedical domain:

> When ***the same MTHC lines*** are exposed to TNF-alpha in combination with IFN-gamma, ***the cells*** instead become DC.

In the above sentence, ***the same MTHC lines*** and ***the cells*** are referring to the same entity and hence are coreferential to each other.

Biomedical research has gained rapid progress over the past few decades. With the advances in biology and life science research, there is a rapidly increasing number of biomedical texts, including research papers, patent documents, and the Web. This results in an increasing demand of applying natural language processing and information retrieval techniques to efficiently exploit text information in large quantities. Lately, biomedical text processing and mining has gained intensive attention in the community of natural language processing and information retrieval. However, coreference resolution, one of the core tasks in natural language processing, only has a small body of prior study in the biomedical domain.

The genre of biomedical texts are mostly scientific writings. Several factors contribute to the differences between coreference resolution in the biomedical domain and in the newswire domain.

For example, biomedical texts use much fewer pronouns than texts in the newswire domain. In fact, the uses of pronouns differ in a variety of domains. Gasperin (2008) pointed out that in her biomedical corpus, Wall Street Journal corpus, and Brown corpus, the percentages of pronouns out of all noun phrases are 3%, 4.5%, and 22%, respectively. Different distribution of the data is known to be a significant reason of different performances when applying the same model to different domains in machine learning. Given that pronoun is one of the important types of noun phrases in coreference resolution, the difference in the use of pronouns contributes significantly to the difference of coreference resolution in different domains.

Another factor that contributes to the domain difference is the use of name acronyms or abbreviations. The alias feature, which utilizes the information of name acronyms or abbreviations, is one of the most important features in coreference resolution. As pointed out by Soon *et al.* (2001), by using only the alias feature, the precisions of coreference resolution on MUC6 and MUC7 data sets are 88.7% and 81.1%, respectively. This suggests that alias is a strong indicator of coreferential relation. Although state-of-the-art named entity recognition achieves good performance in the newswire domain (Finkel *et al.*, 2005), named entity recognition in the biomedical domain is still relatively poor (Zhou *et al.*, 2004). Multiple reasons account for the relatively poor performance. For example, the fact that 62.89% of the words in biomedical named entities are in lowercase makes the initial uppercase feature, one of the important features in named entity recognition, less indicative (Zhou *et al.*, 2004). In coreference resolution in the newswire domain, if one NP is an alias of the other, they are very likely to be coreferential. However, in coreference resolution in the biomedical domain, they are less likely because of the lower performance of named entity recognition. Thus it is possible that two instances with the same feature values have different class labels in different domains. Directly applying a coreference resolution system trained in the newswire domain to the biomedical domain will likely produce errors.

Furthermore, scientific writings in the biomedical domain frequently compare similar objects. For example,

> In Cushing's syndrome, the CR of GR was normal in spite of the fact that the CR of plasma cortisol was disturbed.

The two *CR*s refer to different entities and hence are not coreferential. On the contrary, in the newswire domain, comparisons are less likely, especially for named entities. For example, in the newswire domain, *London* in most cases is coreferential to other *London*s. However, in the biomedical domain, *DNA*s as in *DNA of human beings* and *DNA of monkeys*

are different entities. A coreference resolution system trained from the newswire domain will not be able to capture this difference, such as the two *CR*s or the two *DNA*s in the above examples.

Besides the above, there are other factors which account for the domain differences of coreference resolution in different domains. Due to all these domain differences, directly applying a coreference resolution system trained on a source domain to a different target domain will not achieve good performance. In this thesis, instead of developing specific coreference resolution systems in different domains, we look into a more interesting problem: since we have a coreference resolution system which performs well in the newswire domain, is it possible to adapt the coreference resolution system from the resource-rich newswire domain to another domain, e.g., the biomedical domain, with limited resources but still achieve good performance?

4.1.3 Domain Adaptation for Coreference Resolution

Although coreference resolution systems work well on test texts from the same domain as the training texts, there is a huge performance drop when they are tested on a different domain. This motivates the usage of domain adaptation techniques for coreference resolution: adapting or transferring a coreference resolution system from one source domain that we have a large collection of annotated data, to a second target domain that we want good performance. It is almost inevitable that we annotate *some* data in the target domain to achieve good coreference resolution performance. The question is how to minimize the amount of annotation needed. In the literature, active learning has been exploited to reduce the amount of annotation needed (Lewis and Gale, 1994). In contrast to annotating the entire data set, active learning queries only a subset of the data to annotate in an iterative process. Active learning is a less explored technique in the field of coreference resolution.

Gasperin (2009) tried to apply active learning for anaphora resolution, but found that using active learning was not better than randomly selecting the instances. How to apply active learning, especially integrating it with domain adaptation, remains an open problem for coreference resolution.

The need for coreference resolution on biomedical texts and the small body of prior research on it make the biomedical domain a desirable target domain for evaluating domain adaptation for coreference resolution.

4.2 Domain Adaptation with Active Learning

In this section, we will first present the domain adaptation algorithms and active learning algorithm that we use in our study. Then we show how we combine domain adaptation and active learning together for coreference resolution.

4.2.1 Domain Adaptation

Domain adaptation is applicable when one has a large amount of annotated training data in the source domain and a small amount or none of annotated training data in the target domain. Different domain adaptation techniques have been proposed in the literature. In this thesis, we evaluate the AUGMENT technique introduced by Daume III (2007), as well as the INSTANCE WEIGHTING (IW) and the INSTANCE PRUNING (IP) techniques introduced by Jiang and Zhai (2007).

Augment

Daume III (2007) introduced a very simple but effective domain adaptation technique by feature space augmentation. The motivation of the approach is very intuitive. The source

and the target domains may not be completely different. Otherwise domain adaptation between these two domains may not make sense. Then, there exist some common characteristics that hold in both domains, some characteristics that hold only in the source domain, and some characteristics that hold only in the target domain. The features of each instance we used in the learning-based approaches are in fact capturing the characteristics of that instance. The question is then how to exploit the features to capture the characteristics shared by both domains and the characteristics that hold in each domain.

The AUGMENT technique proposed by Daume III (2007) maps the feature space of each instance into a feature space of higher dimension. Suppose x is the feature vector of an instance. Define Φ^s and Φ^t to be the mappings of an instance from the original feature space to an augmented feature space in the source and the target domain, respectively:

$$\Phi^s(x) = \langle x, x, \mathbf{0} \rangle \tag{4.1}$$

$$\Phi^t(x) = \langle x, \mathbf{0}, x \rangle \tag{4.2}$$

where $\mathbf{0} = \langle 0, 0, \ldots, 0 \rangle$ is a zero vector of length $|x|$. The mapping can be treated as taking each feature in the original feature space and making three versions of it: a general version, a source-specific version, and a target-specific version. The augmented source domain data will contain only the general and the source-specific versions, while the augmented target domain data will contain only the general and the target-specific versions.

Using this augmented feature space, training and testing by learning-based algorithms are the same as using the original feature space. Despite its great simplicity, it achieved good performance in domain adaptation for many natural language processing tasks, e.g. domain adaptation for named entity recognition (Daume III, 2007).

Instance Weighting

Another way to perform domain adaptation is through instance weighting.

Let x and y be the feature vector and the corresponding true label of an instance, respectively. The joint probability $P(x, y)$ will then represent the characteristics and the distribution of the data set. Jiang and Zhai (2007) pointed out that when applying a classifier trained on a source domain to a target domain, the joint probability $P_t(x, y)$ in the target domain may be different from the joint probability $P_s(x, y)$ in the source domain. To conduct domain adaptation, one needs to capture the differences between $P_s(x, y)$ and $P_t(x, y)$. They proposed a general framework to use $P_s(x, y)$ to estimate $P_t(x, y)$.

First of all, the joint probability $P(x, y)$ can be decomposed into two components:

$$P(x, y) = P(y|x)P(x) \tag{4.3}$$

The first component is the conditional probability which captures the probability that an instance x belongs to class y. The second component captures the distribution of the instances in the feature space.

Hence, when conducting domain adaptation, there are also two types of domain differences we need to adapt. The adaptation of the first component is labeling adaptation, while the adaptation of the second component is instance adaptation. The framework proposed by Jiang and Zhai (2007) adapt the differences by instance weighting. In this thesis, we explore only labeling adaptation.

In Section 3.4.1, we have reviewed empirical risk minimization with maximum entropy models. More generally, we want to minimize the expected loss:

$$f^* = \arg\min_{f \in H} \sum_{(x,y) \in X \times Y} P(x, y)L(x, y, f) \tag{4.4}$$

where H is the hypothesis space, x and y are the feature vector and the label of an instance, respectively, X and Y are all the possible values of x and y, respectively, $P(x, y)$ is the joint probability of x and y, and $L(x, y, f)$ is the loss function.

Since we want good performance in the target domain, we want

$$f_t^* = \arg\min_{f \in H} \sum_{(x,y) \in X \times Y} P_t(x, y) L(x, y, f) \tag{4.5}$$

However, we have limited resources in the target domain. Hence, there is not a good estimator of $P_t(x, y)$. But we have plenty of labeled data in the source domain, and hence a good estimator of $P_s(x, y)$. We can rewrite Equation 4.5 as:

$$\begin{aligned} f_t^* &= \arg\min_{f \in H} \sum_{(x,y) \in X \times Y} \frac{P_t(x, y)}{P_s(x, y)} P_s(x, y) L(x, y, f) \\ &= \arg\min_{f \in H} \frac{1}{N_s} \sum_{i=1}^{N_s} \frac{P_t(x_i^s, y_i^s)}{P_s(x_i^s, y_i^s)} L(x_i^s, y_i^s, f) \\ &= \arg\min_{f \in H} \frac{1}{N_s} \sum_{i=1}^{N_s} \frac{P_t(x_i^s) P_t(y_i^s | x_i^s)}{P_s(x_i^s) P_s(y_i^s | x_i^s)} L(x_i^s, y_i^s, f) \\ &= \arg\min_{f \in H} \frac{1}{N_s} \sum_{i=1}^{N_s} \alpha_i \beta_i L(x_i^s, y_i^s, f) \end{aligned} \tag{4.6}$$

where N_s is the number of instances in the source domain, (x_i^s, y_i^s) is the i-th instance in the source domain, and

$$\alpha_i \propto \frac{P_t(x_i^s)}{P_s(x_i^s)} \tag{4.7}$$

$$\beta_i \propto \frac{P_t(y_i^s | x_i^s)}{P_s(y_i^s | x_i^s)} \tag{4.8}$$

In this thesis, we only explore the effect of β_i[1].

Although $P_s(y_i^s|x_i^s)$ can be estimated from the source domain training data, the estimation of $P_t(y_i^s|x_i^s)$ is much harder. Jiang and Zhai (2007) proposed two methods to estimate $P_t(y_i^s|x_i^s)$: INSTANCE WEIGHTING and INSTANCE PRUNING. Both methods first train a classifier with a small amount of target domain training data. Then, INSTANCE WEIGHTING directly estimates $P_t(y_i^s|x_i^s)$ using the trained classifier. INSTANCE PRUNING, on the other hand, removes the top N source domain instances that are predicted wrongly, ranked by the prediction confidence.

Target Domain Instance Weighting

Different from AUGMENT, which conducts domain adaptation in terms of features, INSTANCE WEIGHTING and INSTANCE PRUNING conduct domain adaptation in terms of instances.

Both INSTANCE WEIGHTING and INSTANCE PRUNING set the weights of the source domain instances. In domain adaptation, there are typically many more source domain training instances than target domain training instances. Target domain instance weighting can effectively reduce the imbalance. Unlike INSTANCE WEIGHTING and INSTANCE PRUNING in which each source domain instance is weighted individually, we give all target domain instances the same weight. This target domain instance weighting scheme is not only complementary to INSTANCE WEIGHTING and INSTANCE PRUNING, but is also applicable to AUGMENT. Although target domain instance weighting is also a kind of domain adaptation, we treat it as a separate component.

[1]We have explored the effect of α_i, but the performance was not good.

Putting Things Together

Based on the above analysis, and following the single objective function in Jiang and Zhai (2007), our objective function is

$$f_t^* = \arg\min_{f \in H} [\sum_{i=1}^{N_s} \beta_i L(x_i^s, y_i^s, f) + \lambda_t \sum_{i=1}^{N_t} L(x_i^t, y_i^t, f) + \lambda R(f)] \tag{4.9}$$

where H is the hypothesis space, x_i^s and y_i^s are the feature vector and the label of the i-th instance in the source domain, respectively, x_i^t and y_i^t are the feature vector and the label of the i-th instance in the target domain, respectively, N_s and N_t are the number of instances in the source and the target domain, respectively, β_i is the weight of the i-th instance in the source domain, λ_t is the weight of all instances in the target domain, and $\lambda R(f)$ is a regularization term.

INSTANCE WEIGHTING and INSTANCE PRUNING deal with β_i, while target domain instance weighting dues with λ_t. AUGMENT lifts the feature space to higher dimensions. In this thesis, we explore the effects of these domain adaptation techniques with respect to coreference resolution.

4.2.2 Active Learning

As pointed out in Section 4.1, acquiring labeled data to train supervised learning models for coreference resolution is expensive and time-consuming. Active learning iteratively queries the most informative instances to label, adds them to the training data pool, and trains a new classifier with the enlarged data pool. We follow Lewis and Gale (1994) and use the uncertainty sampling strategy in our active learning setting.

Algorithm 4.1 Algorithm for domain adaptation with active learning

$D_s \leftarrow$ the set of source domain training instances
$D_t \leftarrow$ the set of target domain training instances
$D_a \leftarrow \emptyset$
$\Gamma \leftarrow$ coreference resolution system trained on D_s
$T \leftarrow$ number of iterations
for i from 1 to T **do**
  for each $d_i \in D_t$ **do**
    $\widehat{d_i} \leftarrow$ prediction of d_i using Γ
    $p_i \leftarrow$ confidence of prediction $\widehat{d_i}$
  end for
  $D'_a \leftarrow$ top N instances with the lowest p_i
  $D_a \leftarrow D_a + D'_a$
  $D_t \leftarrow D_t - D'_a$
  provide correct labels to the unlabeled instances in D'_a
  $\Gamma \leftarrow$ coreference resolution system trained on D_s and D_a using the chosen domain adaptation technique
end for

4.2.3 Domain Adaptation with Active Learning

Active learning was originally designed for a single domain. However, it is possible to combine it with domain adaptation. Combining domain adaptation and active learning together, the algorithm is shown in Algorithm 4.1.

In our domain adaptation setting, there is a parameter λ_t for target domain instance weighting. Because the numbers of target domain instances are different in each iteration, the weight should be adjusted in each iteration. We give all target domain training instances an equal weight of $\lambda_t = N_s/N_t$, where N_s and N_t are the numbers of instances in the source domain and the target domain in the current iteration, respectively. We set $N = 10$ to add 10 instances in each iteration to speed up the active learning process.

To provide the correct labels, the labeling process shows the text on the screen, highlights the two NPs, and asks the annotator to decide if they are coreferential. In our experiments, this is simulated by providing the gold annotated coreferential information on this

NP pair to the active learning process.

4.3 Experiments

In this section, we present the experimental settings and the results of domain adaptation with active learning for coreference resolution.

4.3.1 Coreference Resolution System

In the study of domain adaptation for coreference resolution, similar to the experiments in Chapter 3, we use Reconcile, a state-of-the-art coreference resolution system implemented by Stoyanov *et al.* (2009). The input to the coreference resolution system is raw text, and we apply a sequence of preprocessing components to process it. We use the same preprocessing components as described in Section 3.4.1, including: 1) sentence segmentation (using the OpenNLP toolkit); 2) tokenization (using the OpenNLP toolkit); 3) POS tagging (using the OpenNLP toolkit); 4) syntactic parsing (using the Berkeley Parser); and 5) named entity recognition (using the Stanford NER). Markables are extracted as defined in each individual corpus. All possible markable pairs on the training set and the test set are extracted to form training instances and test instances, respectively. The learning algorithm we used is maximum entropy modeling, implemented in the DALR package (Jiang and Zhai, 2007). The feature set we use is RFS, which contains a comprehensive set of 62 features. We do not introduce additional features motivated from the biomedical domain, but use the same feature set for both the source and the target domain. We report coreference resolution results using the MUC evaluation metric.

4.3.2 The Corpora

We explore domain adaptation from the newswire domain to the biomedical domain. The newswire and biomedical domain data that we use are the ACE Phase-2 corpora and the GENIA corpus, respectively. We have described the ACE corpora in Section 3.4.1. The GENIA corpus contains 1,999 MEDLINE abstracts (Yang *et al.*, 2004a,c). We randomly split the GENIA corpus into a training set and a test set, containing 1,600 and 399 texts, respectively.

4.3.3 Preprocessing

For the ACE corpora, all preprocessing components use the original models (provided by the OpenNLP toolkit, the Berkeley Parser, and the Stanford NER). As for the GENIA corpus, since it is from a very different domain, the original models do not perform well. However, the GENIA corpus contains multiple layers of NLP annotations. We use these annotations to re-train each of the preprocessing components using the 1,600 training texts of the GENIA corpus, except tokenization[2]. We do not use any texts in the test set when training these models. Also, we do not use any biomedical domain dependent NLP toolkits, but use general toolkits trained with biomedical training data. These re-trained preprocessing components are then applied to process the entire GENIA corpus, including both the training set and the test set.

Instead of using the entire ACE corpora, we choose the NPAPER portion of the ACE corpora as the source domain in the experiments, because it is the best performing one among the three portions. Under these preprocessing settings and markable extractions, the recall of markables on the training set and the test set of the NPAPER corpus are 94.5%

[2]It turned out that the re-trained tokenization model worked poorer and gave a lot of errors on punctuation symbols. Thus, we stuck to using the original tokenization model.

	NPAPER TRAIN	NPAPER TEST	GENIA TRAIN	GENIA TEST
	# of Docs			
	76	17	1600	399
	# of Words			
Sum	68,463	17,350	391,380	95,405
Avg	900.8	1,020.6	244.6	239.1
	# of Markables			
Sum	21,492	5,153	99,408	24,397
Avg	282.8	303.1	62.1	61.1
	# of Instances			
Sum	3,365,680	871,314	3,335,640	798,844
Avg	44,285.3	51,253.8	2,084.8	2,002.1

Table 4.1: Statistics of the NPAPER and the GENIA corpora

and 95.5%. respectively, while the recall of markables on the training set and the test of the GENIA corpus are 87.6% and 86.6%. respectively. The statistics of the NPAPER and the GENIA corpora are listed in Table 4.1.

4.3.4 Baseline Results

Under the experimental settings, a coreference resolution system that is trained on the NPAPER training set and tested on the NPAPER test set achieves a recall, precision, and F-measure of 59.0%, 70.6%, and 64.3%, respectively, i.e., the NPAPER results of the *All*-style baseline as in Table 3.11. Table 4.2 compares the performance of testing on the GENIA test set, but training with the GENIA training set and the NPAPER training set. Training with in-domain data achieves an F-measure 9.1% higher than training with out-of-domain data. Training with in-domain data is better than training with out-of-domain data in both recall and precision. This confirms the impact of domain difference between newswire domain and biomedical domain for coreference resolution.

Training Set	Recall	Precision	F-measure
GENIA Training Set	37.7	71.9	49.5
NPAPER Training Set	30.3	60.7	40.4

Table 4.2: MUC F-measures on the GENIA test set

4.3.5 Domain Adaptation with Active Learning

In the experiments of domain adaptation with active learning for coreference resolution, we assume that the source domain training data are annotated. The target domain training data are *not* annotated but are used as a data pool for instance selection. The algorithm queries the instances in the data pool to annotate and add them to the training data to update the classifier. The target domain test set is strictly separated from this data pool, i.e., none of the target domain test data are used in the instance selection process of active learning.

From Table 4.1, one can see that both training sets in the NPAPER and the GENIA corpora contain large numbers of training instances. Instead of using the entire training sets in the experiments, we use a smaller subset due to several reasons. First, to train a coreference resolution classifier, we do not need so much training data (Soon *et al.*, 2001). Second, the large number of training instances will slow the active learning process. Third, a smaller source domain training corpus suggests a more modest annotation effort even on the source domain. Lastly, a smaller target domain training corpus means that fewer words need to be read by human annotators to label the data.

We randomly choose 10 NPAPER texts as the source domain training set. A coreference resolution system that is trained on these 10 texts and tested on the entire NPAPER test set achieves a recall, precision, and F-measure of 60.3%, 70.6%, and 65.0%, respectively. This is comparable to (actually slightly better than) a system trained on the entire NPAPER training set. As for the GENIA training set, we randomly choose 40 texts as the target domain training data. To avoid selection bias, we perform 5 random trials, i.e., choosing 5

sets, each containing 40 randomly selected GENIA training texts. In the rest of this chapter, all performances of using *40 GENIA training texts* are the average scores over 5 runs, each of which uses a different set of 40 texts.

In the previous section, we have presented the domain adaptation techniques, the active learning algorithm, as well as the target domain instance weighting scheme. In the rest of this section, we present the experiments to show how domain adaptation, active learning, and target domain instance weighting help coreference resolution in a new domain. We use *Augment*, *IW*, and *IP* to denote the three domain adaptation techniques: AUGMENT, INSTANCE WEIGHTING, and INSTANCE PRUNING, respectively. For a further comparison, we explore another baseline method, which is simply combining the source and the target domain data together, called *Combine* in the rest of this chapter. In all the experiments with active learning, we run 100 iterations, which result in the selection of 1,000 target domain instances.

The first experiment is to measure the effectiveness of target domain instance weighting. We fix on the use of uncertainty based active learning, and compare weighting and without weighting of the target domain instances (denoted as *Weighted* and *Unweighted*). The learning curves for *Combine*, *Augment*, *IW*, and *IP* are shown in Figures 4.1 to 4.4, respectively. For *Combine*, *Augment*, and *IP*, it can be seen that *Weighted* is a clear winner. As for *IW*, at the beginning of active learning, *Unweighted* outperforms *Weighted*, though it is unstable. At the end of the 100 iterations, *Weighted* outperforms *Unweighted*.

Since *Weighted* outperforms *Unweighted*, we fix on the use of *Weighted* and explore the effectiveness of active learning. For comparison, we try another iterative process that randomly selects 10 instances in each iteration. Figures 4.5 to 4.8 show the learning curves of comparing active learning and random selection (denoted as *Uncertainty* and *Random* in the figures, respectively). From the curves, it can be seen that *Uncertainty* outperforms

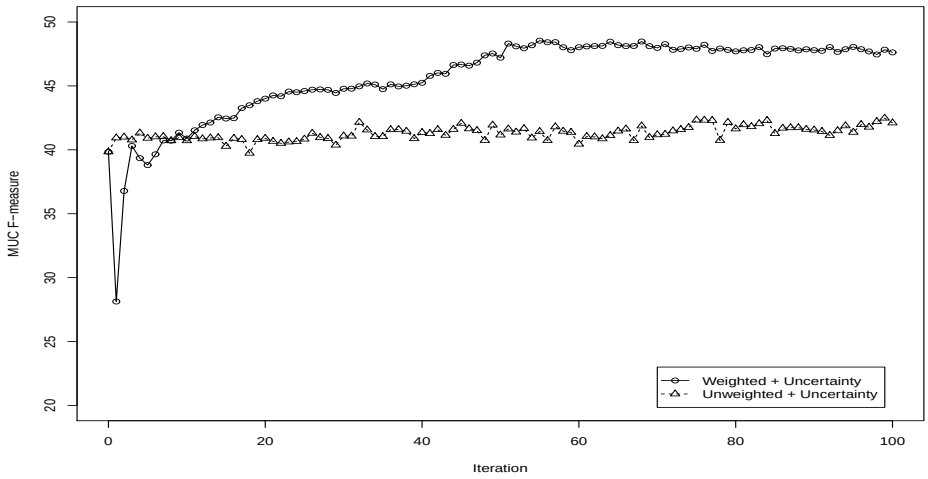

Figure 4.1: Learning curves of comparing target domain instances weighted vs. unweighted. All systems use *Combine* and uncertainty based active learning.

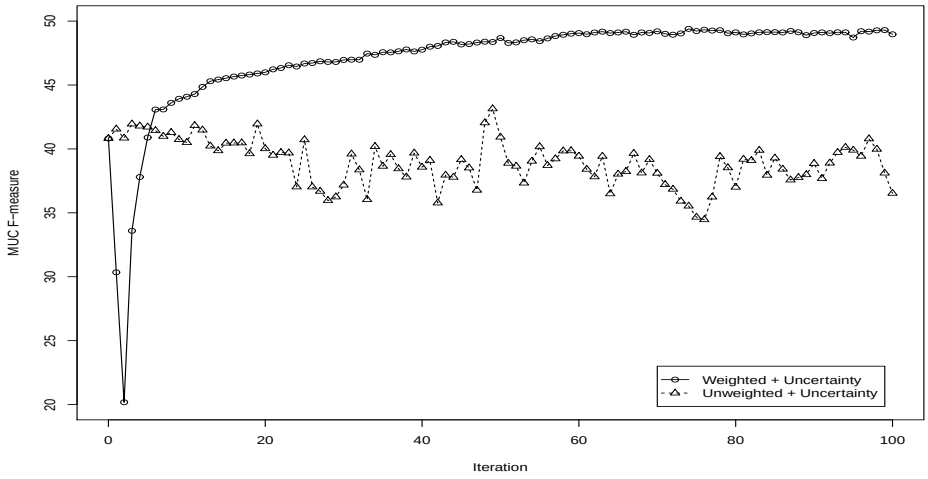

Figure 4.2: Learning curves of comparing target domain instances weighted vs. unweighted. All systems use *Augment* and uncertainty based active learning.

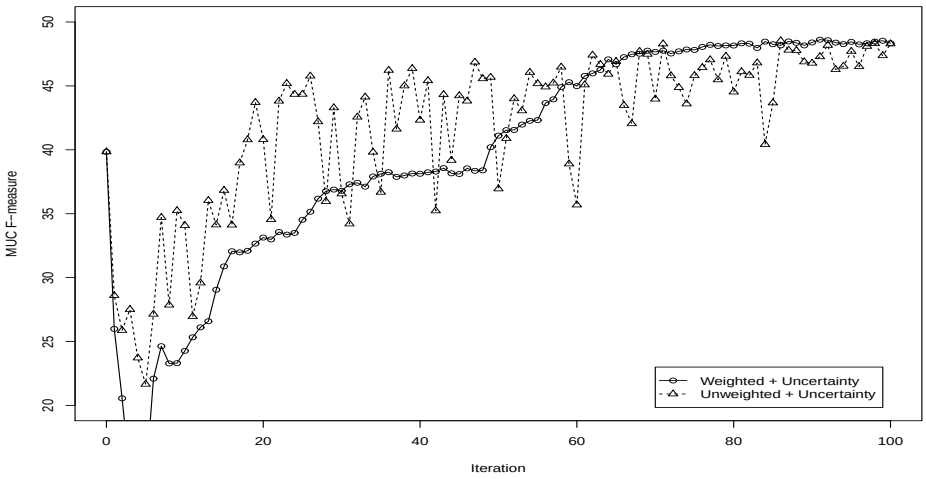

Figure 4.3: Learning curves of comparing target domain instances weighted vs. unweighted. All systems use *IW* and uncertainty based active learning.

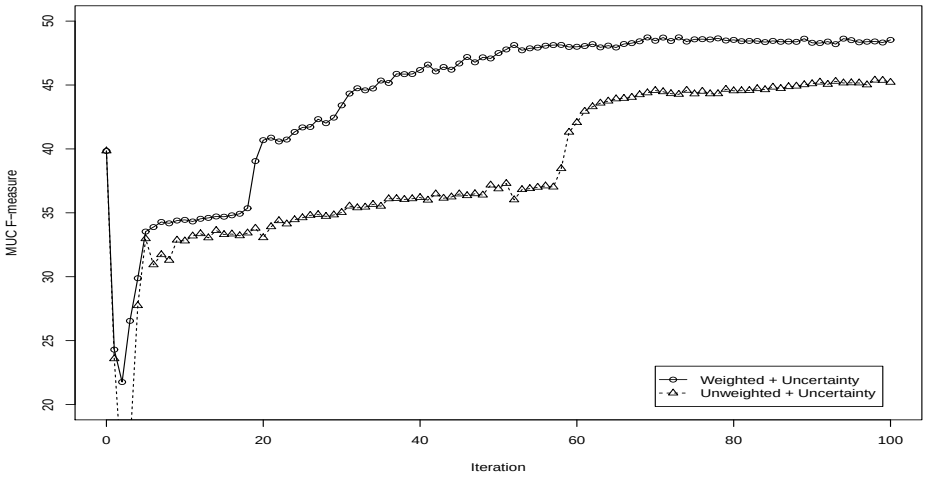

Figure 4.4: Learning curves of comparing target domain instances weighted vs. unweighted. All systems use *IP* and uncertainty based active learning.

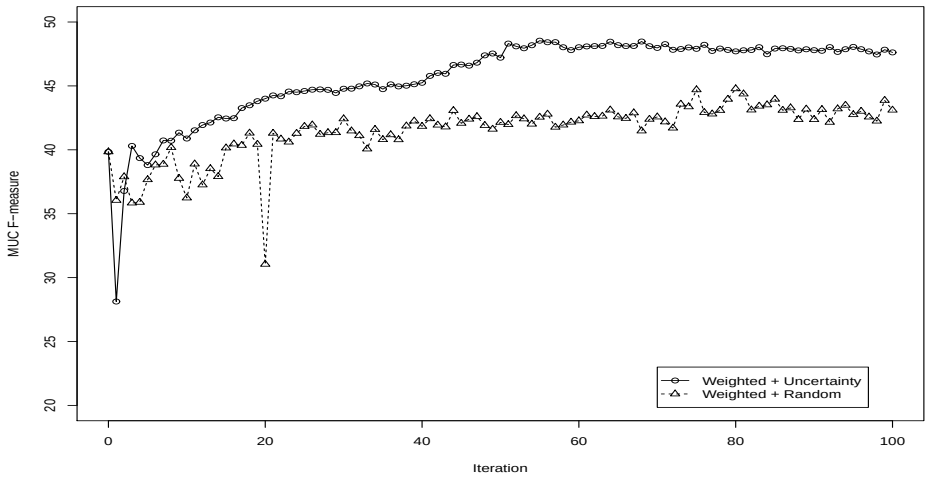

Figure 4.5: Learning curves of comparing uncertainty based active learning vs. random. All systems use *Combine* and *Weighted*.

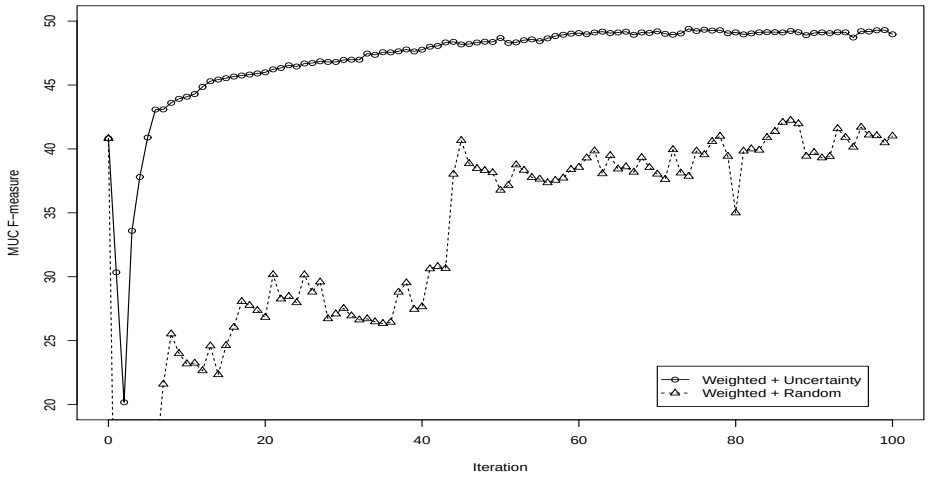

Figure 4.6: Learning curves of comparing uncertainty based active learning vs. random. All systems use *Augment* and *Weighted*.

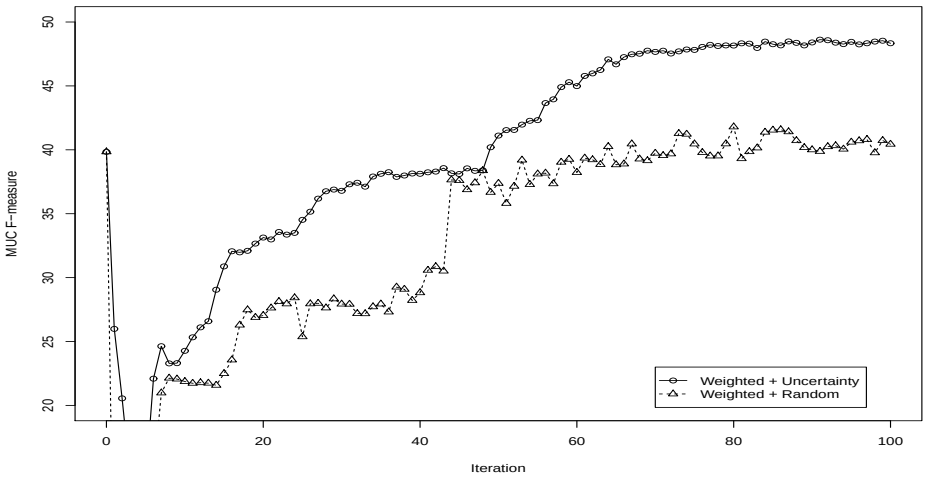

Figure 4.7: Learning curves of comparing uncertainty based active learning vs. random. All systems use *IW* and *Weighted*.

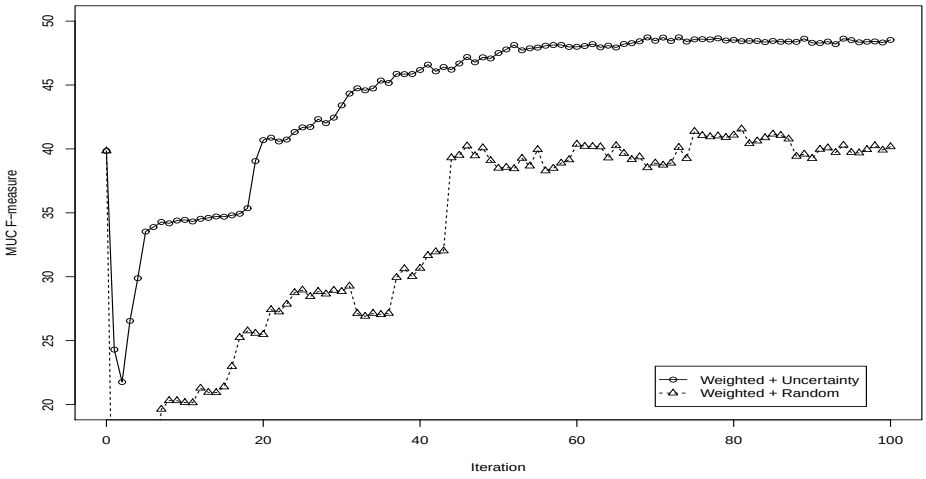

Figure 4.8: Learning curves of comparing uncertainty based active learning vs. random. All systems use *IP* and *Weighted*.

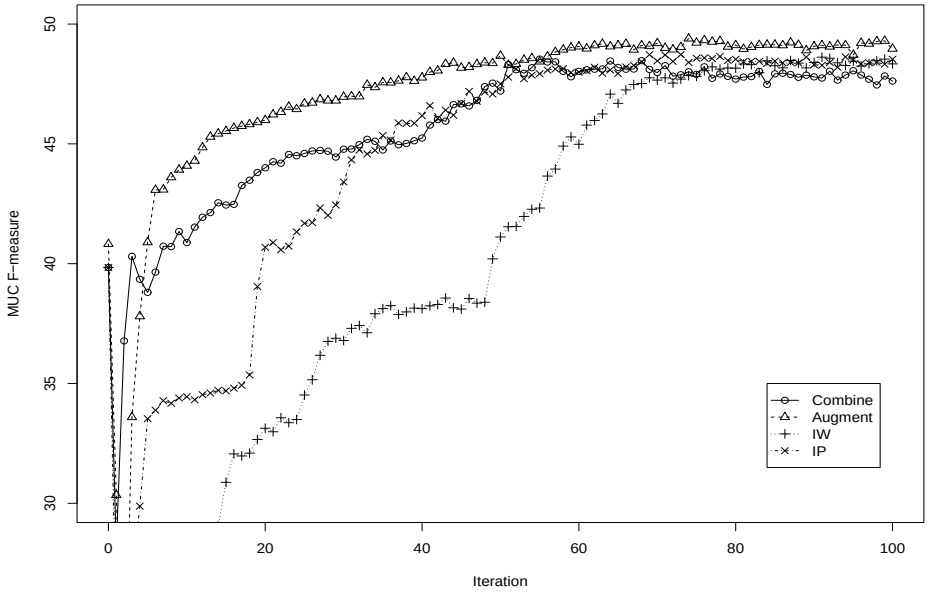

Figure 4.9: Learning curves of different domain adaptation methods. All systems use *Weighted* and *Uncertainty*.

Random in all cases. This is because *Random* may select instances that the classifier has very high confidence in, which will not help to improve the classifier.

In the third experiment, we fix on the use of *Weighted* and *Uncertainty* since they perform the best, and evaluate the effect of different domain adaptation techniques. The learning curves are shown in Figure 4.9. It can be seen that *Augment* is the best performing system. For a closer look, we tabulate the results at every 10 iterations, and list them in Table 4.3 with the statistical significance level indicated.

4.3.6 Analysis

Using only the source domain training data, a coreference resolution system achieves an F-measure of 39.8% on the GENIA test set (the column of "Iteration 0" in Table 4.3). From Figure 4.9 and Table 4.3, we can see that in the first few iterations of active learning, domain

Iteration	0	10	20	30	40	50
Combine+Unweighted	39.8	40.7	40.9	41.1	41.4	41.2
Combine+Weighted	39.8	40.9	44.0**	44.8**	45.2**	47.2**
Augment+Weighted	39.8	**44.1****††	**46.0****††	**47.0****††	**47.8****††	**48.7****††
IW+Weighted	39.8	24.3	33.1	36.8	38.1	41.1
IP+Weighted	39.8	34.4	40.7	43.4**	46.2**††	47.5**
Iteration		60	70	80	90	100
Combine+Unweighted		40.4	41.2	41.6	41.6	42.1
Combine+Weighted		48.0**	48.0**	47.7**	47.8**	47.6**
Augment+Weighted		**49.1****††	**49.2****††	**49.1****††	**49.1****††	**49.0****††
IW+Weighted		45.0**	47.7**	48.2**††	48.4**††	48.3**††
IP+Weighted		48.0**	48.5**††	48.5**††	48.3**††	48.5**††

Table 4.3: MUC F-measures of different active learning settings on the GENIA test set. All systems use *Uncertainty*. Statistical significance is compared against *Combine+Unweighted*, where * and ** stand for $p < 0.05$ and $p < 0.01$, respectively, and against *Combine+Weighted*, where †and ††stand for $p < 0.05$ and $p < 0.01$, respectively.

adaptation does not perform as well as using only the source domain training data. This is because when there are very limited target domain data, the estimation of the target domain is unreliable. Dahlmeier and Ng (2010) reported similar findings though they did not use active learning. With more iterations, i.e., more target domain training data, domain adaptation is clearly superior. Among the three domain adaptation techniques, *Augment* is better than *IW* and *IP*. It not only achieves a higher F-measure, but also a faster speed to adapt to a new domain in active learning. Also, similar to Dahlmeier and Ng (2010), from Table 4.3, we can see that *IP* is generally better than *IW*. All systems with *Weighted* performs much better than *Combine+Unweighted*. This shows the effectiveness of target domain instance weighting. The average recall, precision, and F-measure of our best model, *Augment+Weighted*, after 100 iterations are 37.3%, 71.5%, and 49.0%, respectively. Compared to training with only the NPAPER training data, not only the F-measure, but also both the recall and precision are greatly improved (cf Table 4.2).

Among all the target domain instances that were selected in *Augment+Weighted*, the

average distance of the two markables in an instance (measured in sentence) is 3.4 (averaged over the 5 runs), which means an annotator needs to read at most 4 sentences on average to annotate an instance.

We also investigate the difference of coreference resolution between the newswire domain and the biomedical domain, and the instances that were selected in active learning which represent this difference. As discussed in Section 4.1.2, one of the reasons that coreference resolution differs in the two domains is that the scientific writings of biomedical texts frequently compare entities. For example,

> In Cushing's syndrome, the CR of GR was normal in spite of the fact that the CR of plasma cortisol was disturbed.

The two *CR*s refer to different entities and hence are not coreferential. However, a system trained on NPAPER predicts them as coreferential. In the newswire domain, comparisons are less likely, especially for named entities. A coreference resolution system trained on the newswire domain is unable to capture the difference between these two named entities, hence predicting them as coreferential. For the above sentence, after applying our method, the adapted coreference resolution system is able to predict the two *CR*s as non-coreferential.

Next, we show the effectiveness of our system using domain adaptation with active learning compared to a system trained with full coreferential annotations. Figure 4.10 shows the learning curve of coreference resolution with different sizes of GENIA training texts, when tested on the GENIA test set. Averaged over 5 runs, a system trained on a single GENIA training text achieves an F-measure of 25.9%, which is significantly lower than that achieved by our method. With more GENIA training texts added, the F-measure increases. After 80 texts are used, the system trained on full annotations finally achieves an F-measure of 49.2%, which is 0.2% higher than *Augment+Weighted* after 100 iterations.

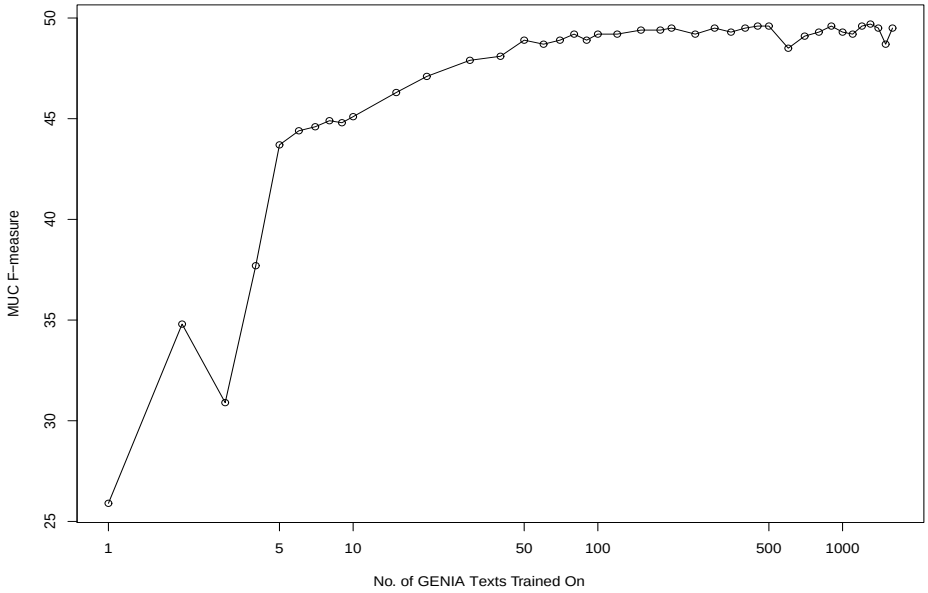

Figure 4.10: The learning curve of coreference resolution with different sizes of GENIA training texts. The F-measures are averaged over 5 runs.

However, after 100 iterations, only 1,000 target domain instances are annotated under our framework. Considering that one single text in the GENIA corpus contains an average of over 2,000 instances (cf Table 4.1), effectively we annotate only half of a text! Compared to the 80 training texts needed, this is a huge reduction. We only need to annotate 1/160 or 0.63% of the training instances under our framework of domain adaptation with active learning.

Lastly, we tabulate the results with the B-CUBED evaluation metric, and list them in Table 4.4 with the statistical significance level indicated. It can be seen that the findings with the MUC evaluation metric can also be seen with the B-CUBED evaluation metric. This suggests that our framework of domain adaptation with active learning for coreference resolution is applicable not only to the MUC evaluation metric, but also to the B-CUBED evaluation metric.

Iteration	0	10	20	30	40	50
Combine+Unweighted	64.5	64.6	64.7	64.7	64.8	64.7
Combine+Weighted	64.5	64.7	66.0**	66.3**	66.4**	67.2**
Augment+Weighted	64.5	**65.8****††	**66.7****††	**67.1****††	**67.4****††	**68.0****††
IW+Weighted	64.5	58.4	61.8	63.2	63.9	64.9*
IP+Weighted	64.5	62.9	65.0*	65.8**	66.9**††	67.4**†
Iteration		60	70	80	90	100
Combine+Unweighted		64.4	64.7	65.0	64.9	65.1
Combine+Weighted		67.6**	67.5**	67.4**	67.4**	67.3**
Augment+Weighted		**68.2****††	**68.3****††	**68.2****††	**68.2****††	**68.2****††
IW+Weighted		66.3**	67.6**	67.7**††	67.9**††	67.8**††
IP+Weighted		67.6**	67.9**††	68.0**††	67.8**††	68.0**††

Table 4.4: B-CUBED F-measures of different active learning settings on the GENIA test set. All systems use *Uncertainty*. Statistical significance is compared against *Combine+Unweighted*, where * and ** stand for $p < 0.05$ and $p < 0.01$, respectively, and against *Combine+Weighted*, where †and ††stand for $p < 0.05$ and $p < 0.01$, respectively.

4.4 Summary

In this chapter, we presented an approach using domain adaptation with active learning to adapt coreference resolution from the newswire domain to the biomedical domain. We explored the effect of domain adaptation, active learning, and target domain instance weighting for coreference resolution. Experimental results showed that domain adaptation with active learning and the target instance weighting scheme achieved a similar performance on MEDLINE abstracts but with a greatly reduced number of annotated training instances, compared to a system trained on full coreference annotations.

Chapter 5

Zero Pronoun Resolution in Chinese

In this chapter, we present a machine learning approach to the identification and resolution of Chinese anaphoric zero pronouns. We perform both identification and resolution automatically, with two sets of easily computable features. Experimental results show that our proposed learning approach achieves anaphoric zero pronoun resolution accuracy comparable to a previous state-of-the-art, heuristic rule-based approach. To our knowledge, our work is the first to perform both identification and resolution of Chinese anaphoric zero pronouns using a machine learning approach.

We start the chapter by first giving a clear task definition of zero pronouns in Chinese and its resolution in Section 5.1. We then give an overview of our approach in Section 5.2. Anaphoric zero pronoun identification and resolution are presented in Section 5.3 and Section 5.4, respectively. We present the experimental results on the blind test set in Section 5.5. Finally, we conclude the chapter in Section 5.6.

5.1 Task Definition

In this section, we clearly explain what a zero pronoun is, define the goal of the task, and describe the data sets and evaluation metrics.

5.1.1 Zero Pronouns

As mentioned in Chapter 1, a zero pronoun (ZP) is a gap (null element) in a sentence which refers to an entity that supplies the necessary information for interpreting the gap. A coreferential zero pronoun is a zero pronoun that is in a coreference relation to one or more overt noun phrases present in the same text.

To facilitate discussion, we reproduce here the example of anaphoric zero pronoun which we have shown in Chapter 1 (originally from the Penn Chinese TreeBank (CTB) (Xue *et al.*, 2005) (sentence ID=300)):

[中国 机电 产品 进出口 贸易]$_1$ 继续 增加 ，
[China electronic products import and export trade]$_1$ continues increasing ,
ϕ_2 占 总 进出口 的 比重 继续 上升 。
ϕ_2 occupies total import and export 's ratio continues increasing .

The anaphoric zero pronoun ϕ_2 refers to the noun phrase 中国机电产品进出口贸易. The corresponding parse tree of the above example is shown in Figure 5.1. In CTB, IP refers to a simple clause that does not have complementizers; CP, on the other hand, refers to a clause introduced by a complementizer.

Just like it is possible that an overt pronoun refers to a non-NP entity, a zero pronoun is not always coreferential with overt noun phrases. Here is an example:

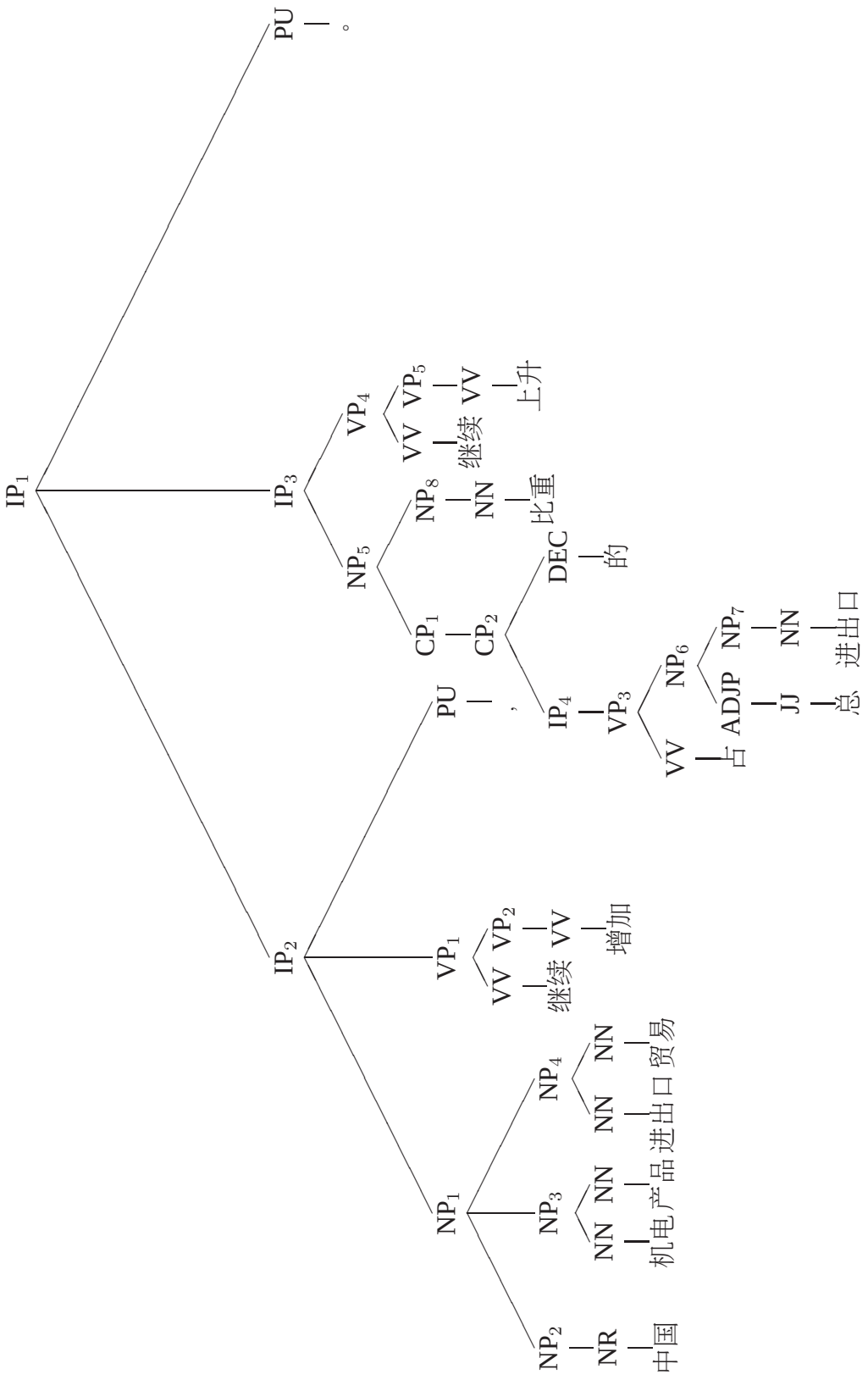

Figure 5.1: The parse tree which corresponds to the anaphoric zero pronoun example in Section 5.1.1.

香港 著名 财团 长江 实业 、 百富勤 作为
Hong Kong famous syndicate Cheung Kong Holdings , Peregrine as
战略性 投资者 已 购入 了 “ 深业 控股 ”
strategic investors already purchased LE “ Shenye Holdings ”
百分之十二 的 股权 , ϕ_3 充分 反映 出 投资者 的
twenty percent 's share , ϕ_3 fully reflects out investors 's
信心 。
confidence .

where the zero pronoun ϕ_3 refers to the event 香港著名财团长江实业、百富勤作为战略性投资者已购入了“深业控股”百分之十二的股权.

As mentioned in Chapter 1, in this thesis, instead of conducting full coreference resolution task for both noun phrases and zero pronouns in Chinese, we focus only on the task of anaphoric zero pronoun identification and resolution, as this is the major difference between coreference resolution in Chinese and in English.

Based on the above definition, the task of zero pronoun resolution is to resolve anaphoric zero pronouns to their correct antecedents. A typical zero pronoun resolution process comprises two stages. The first stage is the identification of the presence of the anaphoric zero pronouns. The second stage is resolving the identified anaphoric zero pronouns to their correct antecedents.

Resolving an anaphoric zero pronoun to its correct antecedent in Chinese is a difficult task. Although gender and number information is available for an overt pronoun and has proven to be useful in pronoun resolution in prior research, a zero pronoun in Chinese, unlike an overt pronoun, provides no such gender or number information. At the same

time, identifying zero pronouns in Chinese is also a difficult task. There are only a few overt pronouns in English, Chinese, and many other languages. State-of-the-art part-of-speech taggers can successfully recognize most of these overt pronouns. However, zero pronouns in Chinese, which are not explicitly marked in a text, are hard to identify. Furthermore, even if a gap is a zero pronoun, it may not be coreferential. All these difficulties make the identification and resolution of anaphoric zero pronouns in Chinese a challenging task.

5.1.2 Corpus

In the study of Chinese anaphoric zero pronouns, we used an annotated third-person pronoun and zero pronoun coreference corpus from Converse (2006)[1]. The corpus contains 205 texts from the Penn Chinese Treebank, with annotations done directly on the parse trees. In the corpus, coreferential zero pronouns, third-person pronouns, and noun phrases are annotated as coreference chains. If a noun phrase is not in any coreference chain, it is not annotated. If a coreference chain does not contain any third-person pronoun or zero pronoun, the whole chain is not annotated.

In the corpus, if a pronoun is not coreferential with any overt noun phrases, it is assigned one of the following six categories:

- discourse deictic (#DD);
- existential (#EXT);
- inferrable (#INFR);
- ambiguity between possible referents in the text (#AMB);
- arbitrary reference (#ARB); and

[1]The data set we obtained is a subset of the one used in Converse (2006).

- unknown (#UNK).

For example, in the second example shown in Section 5.1.1, ϕ_3 refers to an event in the preceding text, with no corresponding antecedent noun phrases. So no antecedent is annotated, and ϕ_3 is labeled as #DD.

Converse (2006) assumed that all correctly identified AZPs and the gold standard parse trees are given as input to her system. She applied the Hobbs algorithm (Hobbs, 1978) to resolve antecedents for the given AZPs.

In our study, we are only interested in zero pronouns with explicit noun phrase referents. If a coreference chain does not contain AZPs, we discard the chain. We also discard the 6 occurrences of zero pronouns with split antecedents, i.e., a zero pronoun with an antecedent that is split into two separate noun phrases. A total of 383 AZPs remain in the data set used in our experiments.

Among the 205 texts in the corpus, texts 1–155 are reserved for training, and the remaining texts (156–205) are used for blind test. The statistics of the training and test data sets are shown in Table 5.1.

	Training	Test
Doc ID	1–155	156–205
# of Texts	155	50
# of Characters	96,338	15,710
# of Words	55,348	9,183
# of ZPs	665	87
# of AZPs	343	40

Table 5.1: Statistics of the corpus for Chinese zero pronouns.

5.1.3 Evaluation Metrics

In the evaluation of identification and resolution of anaphoric zero pronouns in Chinese, we use the same terminology for key and response as in Section 3.1: key is the manually annotated gold standard, and response is the output by an anaphoric zero pronoun identification and resolution system.

Similar to most prior research on pronoun resolution, we evaluate the performance of the identification and resolution of Chinese anaphoric zero pronouns in terms of recall, precision, and F-measure. Similar to the MUC and the B-CUBED metrics we have described in Section 3.1, the overall recall and precision on the test set are computed by micro-averaging over all test instances. The overall F-measure is the F_1-measure, combining both recall and precision, as we have described in Section 3.1.

For AZP identification, recall and precision are defined as:

$$Recall_{AZP} = \frac{\text{\# AZP Hit}}{\text{\# AZP in Key}}$$

$$Precision_{AZP} = \frac{\text{\# AZP Hit}}{\text{\# AZP in Response}}$$

An "AZP Hit" occurs when an AZP as reported in the response has a counterpart in the same position in the gold standard answer key.

For AZP resolution, recall and precision are defined as:

$$Recall_{Resol} = \frac{\text{\# Resol Hit}}{\text{\# AZP in Key}}$$

$$Precision_{Resol} = \frac{\text{\# Resol Hit}}{\text{\# AZP in Response}}$$

A "Resol Hit" occurs when an AZP is correctly identified, and it is correctly resolved to a noun phrase that is in the same coreference chain as provided in the answer key.

5.2 Overview of Our Approach

In this section, we give an overview of our approach for Chinese AZP identification and resolution, as well as the experimental settings.

Similar to coreference resolution in English as described in Section 3.4, we need to process the input raw texts before resolving the zero pronouns: the raw texts need to be processed by a Chinese word segmenter, a part-of-speech (POS) tagger, and a parser sequentially. Although our approach can apply directly to machine-generated parse trees from raw text, in order to minimize errors introduced by preprocessing, and focus mainly on Chinese zero pronoun resolution, we used the gold standard word segmentation, POS tags, and parse trees provided by CTB in our experiments. However, we removed all null categories and functional tags from the CTB gold standard parse trees. Figure 5.1 shows a parse tree after such removal.

A set of zero pronoun candidates and a set of noun phrase candidates are then extracted. If W is the leftmost word in the word sequence that is spanned by some VP node, the gap G that is immediately to the left of W qualifies as a ZP candidate. For example, in Figure 5.1, gaps immediately to the left of the two occurrences of 继续, and 增加, 占, 上升 are all ZP candidates. All noun phrases[2] that are either maximal NPs or modifier NPs qualify as NP candidates. For example, in Figure 5.1, NP_1, NP_2, NP_3, NP_5, and NP_6 are all NP candidates. With these ZP and NP candidate extractions, the recall percentages of ZPs on the training and test data sets were both 100%, and the recall percentages of NPs on the

[2] A noun phrase can either be NP or QP (a number or a quantifier) in the CTB. We simply use NP hereafter.

training and test data sets were 98.6% and 99.0%, respectively.

After the ZP and NP candidates are determined, we perform AZP identification and resolution in a sequential manner. We build two classifiers, the AZP identification classifier and the AZP resolution classifier. The AZP identification classifier determines the positions of AZPs, while the AZP resolution classifier finds an antecedent noun phrase for each AZP identified by the AZP identification classifier. Both classifiers are built using machine learning techniques. The features of both classifiers are largely syntactic features based on parse trees and are easily computed.

In the experiments, we performed 5-fold cross validation on the training data set to tune parameters and to pick the best model. We then retrained the best model with all data in the training data set, and applied it to the blind test set. In the following sections, all accuracies reported on the training data set are based on 5-fold cross validation.

5.3 Anaphoric Zero Pronoun Identification

In this section, we describe how we identify the anaphoric zero pronouns, and present the experimental results.

5.3.1 The Features

We use machine learning techniques to build the AZP identification classifier. Each training or test instance is formed by a ZP candidate. A set of features is used for the learning-based approach and is extracted for each instance. For easy description of the features, we introduce some notations before describing the features.

Let Z be a ZP candidate, W_l and W_r be the words immediately to the left and to the right of Z, respectively, P the parse tree node that is the lowest common ancestor node

of W_l and W_r, P_l and P_r the child nodes of P that are ancestor nodes of W_l and W_r, respectively. If Z is the first gap of the sentence, the values of W_l, P, P_l, and P_r are all *NA*. Furthermore, let V be the highest VP node in the parse tree that is immediately to the right of Z, i.e., the leftmost word in the word sequence that is spanned by V is W_r. If Z is not the first gap in the sentence, define the ceiling node C to be P, otherwise to be the root node of the parse tree. In the example shown in Figure 5.1, for the ZP candidate ϕ_2 (which is immediately to the left of 占), W_l, W_r, P, P_l, P_r, V, and C are "，", 占, IP_1, IP_2, IP_3, VP_3, and IP_1, respectively.

The list of features is given below. For easy understanding, the feature values of the ZP candidate ϕ_2 in Figure 5.1 are given as an example after the description of each feature.

1. First_Gap: If Z is the first gap in the sentence, T; else F (ϕ_2:F)

2. P_l_Is_NP: If Z is the first gap in the sentence, NA; otherwise, if P_l is an NP node, T; else F. (ϕ_2:F)

3. P_r_Is_VP: If Z is the first gap in the sentence, NA; otherwise, if P_r is a VP node, T; else F. (ϕ_2:F)

4. P_l_Is_NP & P_r_Is_VP: If Z is the first gap in the sentence, NA; otherwise, if P_l is an NP node and P_r is a VP node, T; else F. (ϕ_2:F)

5. P_Is_VP: If Z is the first gap in the sentence, NA; otherwise, if P is a VP node, T; else F. (ϕ_2:F)

6. IP-VP: If in the path from W_r to C, there is a VP node such that its parent node is an IP node, T; else F. (ϕ_2:T)

7. Has_Ancestor_NP: If V has an NP node as ancestor, T; else F. (ϕ_2:T)

Model	R	P	F
Heuristic	99.7	15.0	26.1
AZP Ident	19.8	51.1	28.6
AZP Ident ($r = 8$)	59.8	44.3	50.9

Table 5.2: Results of AZP identification on the training data set under 5-fold cross validation.

8. Has_Ancestor_VP: If V has a VP node as ancestor, T; else F. (ϕ_2:F)

9. Has_Ancestor_CP: If V has a CP node as ancestor, T; else F. (ϕ_2:T)

10. Left_Comma: If Z is the first gap, NA; otherwise if W_l is a comma, T; else F. (ϕ_2:T)

11. Subject_Role: If the grammatical role of Z is subject, S; else X. (ϕ_2:X)

12. Clause: If V is in a matrix clause, an independent clause, a subordinate clause, or none of the above, the value is M, I, S, X, respectively. (ϕ_2:I)

13. Is_In_Headline: If Z is in the headline of the text, T; else F. (ϕ_2:F)

5.3.2 Training and Testing

To train an AZP identification classifier, we generate training instances from the training data set. Each ZP candidate we have extracted from the training data set forms one training instance. A training instance is positive if the ZP candidate is an AZP, and negative if it is not.

After generating all training instances, we train an AZP identification classifier using J48, the WEKA implementation of the C4.5 decision tree. During testing, each ZP candidate is presented to the learned classifier to determine whether it is an AZP. The experimental results of AZP identification of 5-fold cross validation on the training data set are shown in the row "AZP Ident" in Table 5.2. It achieved an F-measure of 28.6%.

We use heuristic rules as a baseline for comparison. The rules used by the heuristic model are as follows. For a node T in the parse tree, if

1. T is a VP node; and
2. T's parent node is not a VP node; and
3. T has no left sibling, or its left sibling is not an NP node,

then the gap that is immediately to the left of the word sequence spanned by T is an AZP. The results of the heuristic baseline are shown in the row "Heuristic" in Table 5.2. This simple AZP identification heuristic achieved an F-measure of 26.1%.

5.3.3 Imbalanced Training Data

From Table 5.2, one can see that the F-measure of the machine-learned AZP identification model is 28.6%, which is only slightly higher than the baseline heuristic model. It has a relatively high precision, but much lower recall. The problem lies in the highly imbalanced number of positive and negative training instances. Among all the 155 texts in the training set, there are 343 positive and 10,098 negative training instances. The ratio r of the number of negative training instances to the number of positive training instances is 29.4. A classifier trained on such highly imbalanced training instances tends to predict more test instances as negative instances. This explains why the precision is high, but the recall is low.

To overcome this problem, we vary r by varying the weights of the positive training instances, which is equivalent to sampling more positive training instances. The values of r that we have tried are $1, 2, 3, \ldots, 29$. The larger the value of r, the higher the precision, and the lower the recall. By tuning r, we get a balance between precision and recall, and

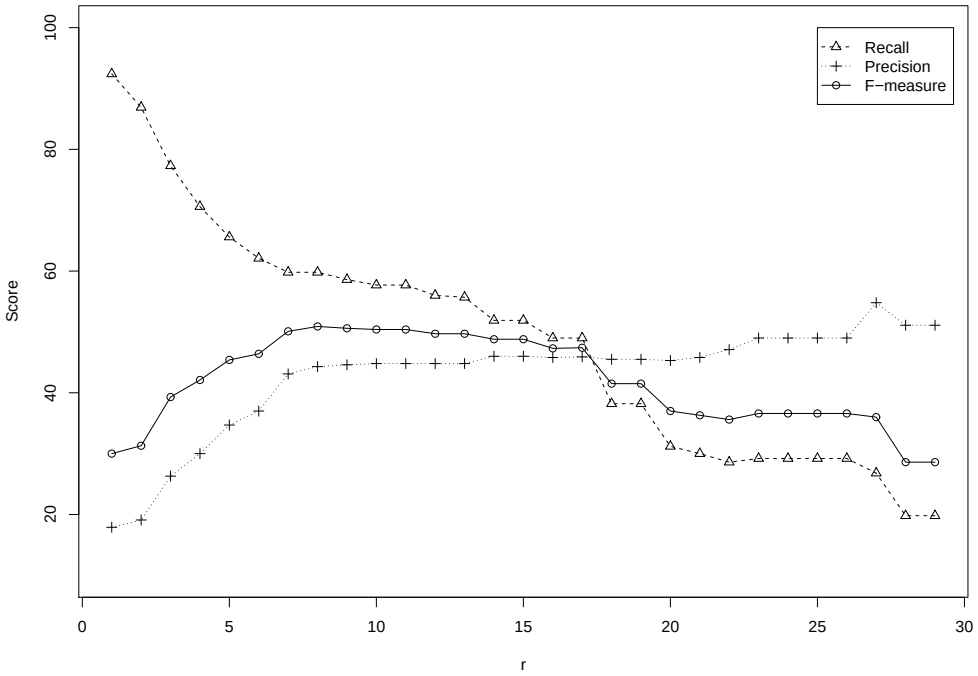

Figure 5.2: Effect of tuning r on AZP identification (the default r in our dataset is 29.4)

hence an optimal F-measure. Figure 5.2 shows the effect of tuning r on AZP identification. When $r = 8$, the optimal F-measure is 50.9%, which is much higher than the F-measure without tuning r. The result is shown in the row "AZP Ident ($r = 8$)" in Table 5.2.

Ng and Cardie (2002b) reported that the accuracies of their noun phrase anaphoricity determination classifier were 86.1% and 84.0% on the MUC6 and the MUC7 data sets, respectively. Noun phrases provide much fruitful information for anaphoricity identification. However, useful information such as gender, number, lexical string, etc, is not available in the case of zero pronouns. This makes AZP identification a much more challenging task, and hence it has a relatively low accuracy.

5.4 Anaphoric Zero Pronoun Resolution

In this section, we describe how we resolve the identified anaphoric zero pronouns to their NP antecedents, and present the experimental results.

5.4.1 The Features

Similar to AZP identification, we also use machine learning techniques to build a classifier for AZP resolution. An instance is an NP–ZP pair. A set of features is used for the learning-based approach. Again, let Z be the anaphoric zero pronoun that is under consideration, and A be the potential NP antecedent for Z. V is the same as in AZP identification, i.e., the highest VP node in the parse tree that is immediately to the right of Z.

The features for learning the classifier for anaphoric zero pronoun resolution are described as follows. For easy understanding, the feature values of the pair of the potential NP antecedent and the ZP candidate NP$_1$–ϕ_2 in Figure 5.1 are given as an example after the description of each feature.

- Features between Z and A

  1. Dist_Sentence: If Z and A are in the same sentence, 0; if they are one sentence apart, 1; and so on. (NP$_1$–ϕ_2:0)
  2. Dist_Segment: If Z and A are in the same segment (where a segment is a sequence of words separated by punctuation marks including "，", "；", "。", "！", and "？"), 0; if they are one segment apart, 1; and so on. (NP$_1$–ϕ_2:1)
  3. Sibling_NP_VP: If Z and A are in different sentences, F; Otherwise, if both A and Z are child nodes of the root node, and they are siblings (or at most separated by one comma), T; else F. (NP$_1$–ϕ_2:F)

4. Closest_NP: If A is the closest preceding NP candidate to Z, T; else F. (NP_1–ϕ_2:T)

- Features on A

5. A_Has_Anc_NP: If A has an ancestor NP node, T; else F. (NP_1–ϕ_2:F)
6. A_Has_Anc_NP_In_IP: If A has an ancestor NP node which is a descendant of A's lowest ancestor IP node, T; else F. (NP_1–ϕ_2:F)
7. A_Has_Anc_VP: If A has an ancestor VP node, T; else F. (NP_1–ϕ_2:F)
8. A_Has_Anc_VP_In_IP: If A has an ancestor VP node which is a descendant of A's lowest ancestor IP node, T; else F. (NP_1–ϕ_2:F)
9. A_Has_Anc_CP: If A has an ancestor CP node, T; else F. (NP_1–ϕ_2:F)
10. A_Grammatical_Role: If the grammatical role of A is subject, object, or others, the value is S, O, or X, respectively. (NP_1–ϕ_2:S)
11. A_Clause: If A is in a matrix clause, an independent clause, a subordinate clause, or none of the above, the value is M, I, S, X, respectively. (NP_1–ϕ_2:M)
12. A_Is_ADV: If A is an adverbial NP, T; else F. (NP_1–ϕ_2:F)
13. A_Is_TMP: If A is a temporal NP, T; else F. (NP_1–ϕ_2:F)
14. A_Is_Pronoun: If A is a pronoun, T; else F. (NP_1–ϕ_2:F)
15. A_Is_NE: If A is a named entity, T; else F. (NP_1–ϕ_2:F)
16. A_In_Headline: If A is in the headline of the text, T; else F. (NP_1–ϕ_2:F)

- Features on Z

17. Z_Has_Anc_NP: If V has an ancestor NP node, T; else F. (NP_1–ϕ_2:T)

18. Z_Has_Anc_NP_In_IP: If V has an ancestor NP node which is a descendant of V's lowest ancestor IP node, T; else F. (NP_1–ϕ_2:F)
19. Z_Has_Anc_VP: If V has an ancestor VP node, T; else F. (NP_1–ϕ_2:F)
20. Z_Has_Anc_VP_In_IP: If V has an ancestor VP node which is a descendant of V's lowest ancestor IP node, T; else F. (NP_1–ϕ_2:F)
21. Z_Has_Anc_CP: If V has an ancestor CP node, T; else F. (NP_1–ϕ_2:T)
22. Z_Grammatical_Role: If the grammatical role of Z is subject, S; else X. (NP_1–ϕ_2:X)
23. Z_Clause: If V is in a matrix clause, an independent clause, a subordinate clause, or none of the above, the value is M, I, S, X, respectively. (NP_1–ϕ_2:I)
24. Z_Is_First_ZP: If Z is the first ZP candidate in the sentence, T; else F. (NP_1–ϕ_2:F)
25. Z_Is_Last_ZP: If Z is the last ZP candidate in the sentence, T; else F. (NP_1–ϕ_2:F)
26. Z_In_Headline: If Z is in the headline of the text, T; else F. (NP_1–ϕ_2:F)

5.4.2 Training and Testing

To train an AZP resolution classifier, we generate training instances from the training data set. We generate training instances in the following way. An AZP Z and its immediately preceding coreferential NP antecedent A in the gold standard coreference chain form a positive training instance. Between A and Z, there are other NP candidates. Each one of these NP candidates, together with Z, form a negative training instance. This is similar to the approach adopted in Soon *et al.* (2001). We also train the AZP resolution classifier using the J48 decision tree learning algorithm.

After building both AZP identification and resolution classifiers, we perform AZP identification and resolution in a sequential manner. For a ZP candidate Z, the AZP identification classifier determines whether Z is an AZP. If it is an AZP, all NP candidates that are to the left of Z in textual order are considered as potential antecedents. These potential antecedents are tested from back to front. We start from the NP candidate A_1 that is immediately to the left of Z. A_1 and Z form a pair. If the pair is classified as positive by the resolution classifier, A_1 is the antecedent for Z. If it is classified as negative, we proceed to the NP candidate A_2 that is immediately to the left of A_1, and test again. The process continues until we find an antecedent for Z, or there is no more NP candidate to test.

This right-to-left search attempts to find the closest correct antecedent for an AZP. We do not choose the best-first search strategy proposed by Ng and Cardie (2002c). This is because we generate training instances and build the resolution classifier by pairing each zero pronoun with its closest preceding antecedent. In addition, a zero pronoun is typically not too far away from its antecedent. In our data set, 92.6% of the AZPs have antecedents that are at most 2 sentences apart. Our experiment shows that this closest-first strategy performs better than the best-first strategy for Chinese AZP resolution.

Table 5.3 shows the experimental results of 5-fold cross validation on the training data set. The results of AZP identification followed by resolution are shown in the row "AZP Ident (r=8 t=0.5)" in the table. We achieved an F-measure of 20.1%. For comparison, we show three baseline systems. In all three baseline systems, we do not perform AZP identification, but directly apply the AZP resolution classifier. In the first baseline, we apply the AZP resolution classifier on all ZP candidates. In the second baseline, we apply the classifier only on ZPs annotated in the gold standard, instead of all ZP candidates. In the third baseline, we further restrict it to resolve only AZPs. The results of the three baselines are reported in the rows "All ZP Candidates", "Gold ZP", and "Gold AZP", respectively, in

Model	R	P	F
All ZP Candidates	40.5	1.3	2.5
Gold ZP	40.5	20.9	27.6
Gold AZP	40.5	40.6	40.6
AZP Ident (r=8 t=0.5)	23.6	17.5	20.1
AZP Ident (r=11 t=0.6)	22.4	20.3	21.3

Table 5.3: Results of AZP resolution on the training data set under 5-fold cross validation.

Table 5.3. The F-measures of the three baselines are 2.5%, 27.6%, and 40.6%, respectively.

5.4.3 Tuning of Parameters

Ng (2004b) showed that an NP anaphoricity identification classifier with a cut-off threshold $t = 0.5$ pruned away many correct anaphoric NPs and harmed the overall recall. By varying t, the overall resolution F-measure was improved. We adopt the same tuning strategy and accept a ZP candidate ZP_i as an AZP and proceed to find its antecedent only if $P(ZP_i) \geq t$. The possible values for t that we have tried are $0, 0.05, 0.1, \ldots, 0.95$.

In Section 5.3, we have shown that $r = 8$ yields the best AZP identification F-measure. When we fix $r = 8$ and vary t, the overall F-measure for AZP resolution is the best at $t = 0.65$, as shown in Figure 5.3. We then try tuning r and t at the same time. An overall optimal F-measure of 21.3% is obtained when $r = 11$ and $t = 0.6$. The results are shown in the row "AZP Ident (r=11 t=0.6)" in Table 5.3. We compare this tuned F-measure with the F-measure of 20.1% when $r = 8$ and $t = 0.5$, obtained without tuning t. Although the improvement is modest, it is statistically significant ($p < 0.05$).

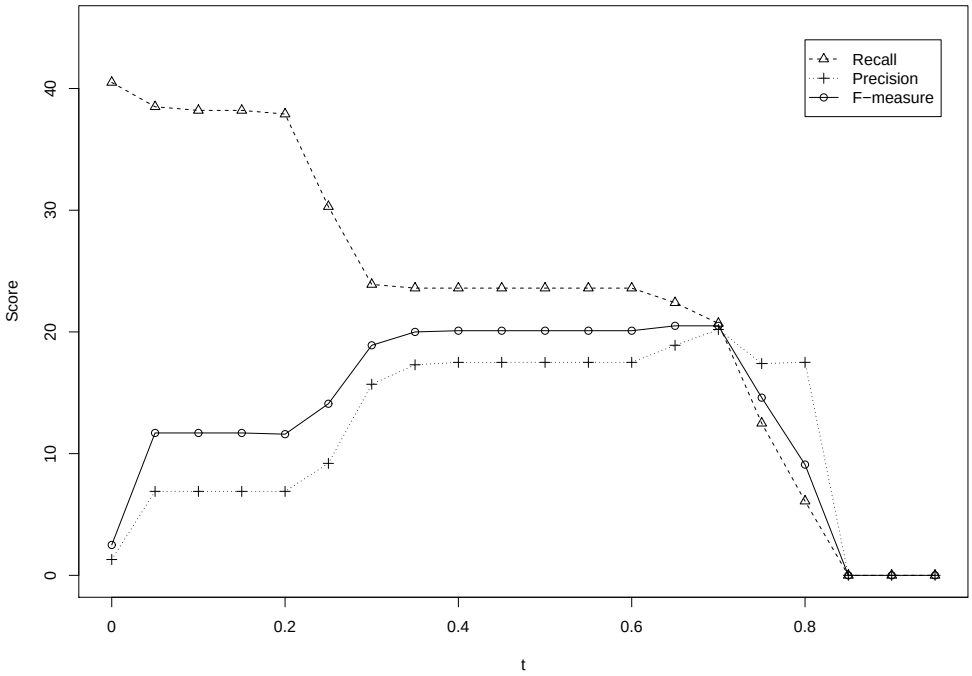

Figure 5.3: Effect of tuning t on AZP resolution

5.5 Experimental Results

In the previous section, we have shown that when $r = 11$ and $t = 0.6$, our sequential AZP identification and resolution achieves the best F-measure under 5-fold cross validation on the 155 training texts. In order to utilize all available training data, we generate training instances for the AZP identification classifier with $r = 11$, and generate training instances for the AZP resolution classifier, on all 155 training texts. Both classifiers are trained again with the newly generated training instances. We then apply both classifiers with anaphoricity identification cut-off threshold $t = 0.6$ to the blind test data. The results are shown in Table 5.4. We achieved an F-measure of 25.9% on the blind test set.

R	P	F
27.5	24.4	25.9

Table 5.4: Results of AZP resolution on blind test data.

Converse (2006) assumed all AZPs are given and correctly input to her system. She found an antecedent for each known AZP by utilizing all available information on the gold standard parse trees. The accuracy of her rule-based approach was 43.0%. As a comparison, given gold standard AZPs, under 5-fold cross validation of all 205 texts in the corpus, our system achieved recall, precision, and F-measure of 42.3%, 42.7%, and 42.5%, respectively. This shows that our proposed machine learning approach for Chinese zero pronoun resolution is comparable to her rule-based approach.

5.6 Summary

In this chapter, we presented a machine learning approach to the identification and resolution of Chinese anaphoric zero pronouns. We performed both identification and resolution automatically, with two sets of easily computable features. Experimental results showed that our proposed learning approach achieved anaphoric zero pronoun resolution accuracy comparable to a previous state-of-the-art, heuristic rule-based approach. To our knowledge, our work is the first to perform both identification and resolution of Chinese anaphoric zero pronouns using a machine learning approach.

Chapter 6

Conclusion

In this thesis, we presented a novel maximum metric score training approach comprising the use of instance weighting and beam search to maximize the chosen coreference metric score on the training corpus during training. We also explored the integration of domain adaptation and active learning for coreference resolution from a newswire source domain where we have a large collection of annotated data, to a second biomedical target domain in which we want good performance. Lastly, we presented the first machine learning approach to the identification and resolution of Chinese anaphoric zero pronouns. This chapter summarizes our work and outlines possible future research directions.

6.1 Summary

Most previous work on coreference resolution either failed to maximize the evaluation metric score of coreference resolution, or maximized it during testing. Typically, during training, a coreference resolution system minimizes the number of misclassified training instances without considering the evaluation metric. However, the extracted training instances are not only not equally easy to be classified, but also not equally important. To

address this deficiency, we proposed MMST, a generic framework to train a classifier to maximize the chosen metric score for coreference resolution by iteratively assigning higher weights to the hard-to-classify training instances. Experimental results showed that MMST achieved statistically significant improvements over the *Soon*-style and the *All*-style baselines on all the five standard benchmark corpora (two MUC corpora and three ACE corpora), with both the link-based MUC metric and the mention-based B-CUBED metric.

One of the most challenging obstacles in applying supervised learning approaches to coreference resolution is the difficulty of data annotation. It is much more time-consuming and expensive to annotate a corpus for coreference resolution than to annotate a corpus for other natural language processing tasks. To achieve good coreference resolution performance in a new domain, it is almost inevitable that we annotate some data. This raises the question of how to minimize the amount of data annotation needed while maintaining good coreference resolution performance. In this thesis, we presented an approach comprising domain adaptation and active learning together to adapt coreference resolution from the newswire source domain to the biomedical target domain. We explored the effect of domain adaptation, active learning, and target domain instance weighting for coreference resolution. Experimental results showed that domain adaptation with active learning and the target instance weighting scheme achieved a similar performance on MEDLINE abstracts, but with a greatly reduced number of training instances that we need to annotate, compared to a system trained on full coreference annotations.

There exist some language-specific linguistic phenomena which make coreference resolution in one language different from the others. Zero pronouns, one of these phenomena, occur much more frequently in Chinese than in English, and pose a unique challenge for coreference resolution in Chinese. Although Chinese zero pronouns have been studied

from the perspective of linguistics, only a small body of prior research studied this phenomenon from the perspective of computational linguistics. All previous research on zero pronoun identification and resolution in Chinese uses hand-engineered rules or heuristics. In this thesis, we presented a machine learning approach to the identification and resolution of Chinese anaphoric zero pronouns. We performed both identification and resolution automatically, with two sets of easily computable features. Experimental results showed that our proposed learning approach achieved anaphoric zero pronoun resolution accuracy comparable to a previous state-of-the-art, heuristic rule-based approach. To our knowledge, our work is the first to perform both identification and resolution of Chinese anaphoric zero pronouns using a machine learning approach.

6.2 Future Directions

There are numerous avenues to extend the current work. In this section, we discuss some of these possibilities.

Many natural language processing tasks benefit from coreference resolution. However, recall and precision for coreference resolution may not have equal importance for all tasks. Furthermore, not all noun phrase types are equally important in different tasks, too. For example, in question answering, resolving a pronoun to its correct antecedent may be more critical than resolving a named entity to its correct antecedent. Directly applying a coreference resolution system with the highest MUC or B-CUBED metric to question answering may be sub-optimal. In this thesis, we have evaluated the MUC and the B-CUBED metrics in standard benchmark corpora. Under the MMST framework, it is possible to apply our method to other scenarios by simply replacing the evaluation metric with the desired

ones. Exploring alternative evaluation metrics of coreference resolution for different applications in NLP and IR, and adopting our MMST framework to maximize the contribution of coreference resolution to these tasks are interesting directions to pursue in the future.

It can also be seen that in the beam search algorithm, there are potentially other ways of updating the weights. In this thesis, we explored the weight updating method in the beam search algorithm by differentiating false positive and false negative training instances. There are potentially other ways to update the weights, e.g., setting the weight of an instance to be proportional to its impact on the chosen evaluation metric score.

To adapt a coreference resolution system to a new domain, we have explored the integration of domain adaptation and active learning to greatly reduce the number of instances we need to annotate in the desired target domain. In our active learning setting, we followed Lewis and Gale (1994) and used the uncertainty sampling strategy. As pointed out by prior work, compared to uncertainty based sampling, density based sampling (Cohn *et al.*, 1996) has the potential to perform better when very few iterations in active learning have been conducted, because it samples from dense unlabeled regions and picks the instances that affect the most remaining unlabeled data. The DUAL algorithm, which combines uncertainty and density based sampling, may improve it even further (Donmez *et al.*, 2007).

Our work is the first to perform both identification and resolution of Chinese anaphoric zero pronouns using a machine learning approach. Given that the performance of our zero pronoun identification and resolution system is still modest, there is much room for improvement in both the identification and the resolution part. Besides, applying zero pronoun identification and resolution directly on machine-generated parse trees need to be investigated. Finally, applying zero pronoun identification and resolution in Chinese to other natural language processing tasks, e.g., machine translation, is one of the fruitful areas for future research.

Bibliography

Aone, Chinatsu and Scott William Bennett (1995). Evaluating automated and manual acquisition of anaphora resolution strategies. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL1995)*, pages 122–129, Cambridge, Massachusetts, USA.

Bagga, Amit and Breck Baldwin (1998). Algorithms for scoring coreference chains. In *Proceedings of the 1st International Conference on Language Resources and Evaluation (LREC1998)*, pages 563–566, Granada, Spain.

Barzilay, Regina and Mirella Lapata (2005). Modeling local coherence: An entity-based approach. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL2005)*, pages 141–148, Ann Arbor, Michigan, USA.

Berger, Adam L., Vincent J. Della Pietra, and Stephen A. Della Pietra (1996). A maximum entropy approach to natural language processing. *Computational Linguistics*, **22**(1), 39–71.

Bergler, Sabine, René Witte, Michelle Khalife, Zhuoyan Li, and Frank Rudzicz (2003). Using knowledge-poor coreference resolution for text summarization. In *Proceedings of the DUC2003 Workshop on Text Summarization*, Edmonton, Canada.

Bergsma, Shane and Dekang Lin (2006). Bootstrapping path-based pronoun resolution. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL2006)*, pages 33–40, Sydney, Australia.

Castaño, José, Jason Zhang, and James Pustejovsky (2002). Anaphora resolution in biomedical literature. In *International Symposium on Reference Resolution*.

Chan, Yee Seng and Hwee Tou Ng (2007). Domain adaptation with active learning for word sense disambiguation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL2007)*, pages 49–56, Prague, Czech Republic.

Chinchor, Nancy (1995). Statistical significance of MUC-6 results. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pages 39–43, Columbia, Maryland, USA.

Chomsky, Noam (1981). *Lectures on Government and Binding*. Foris, Dordrecht.

Cohen, K. Bretonnel, Arrick Lanfranchi, William Corvey, William A. Baumgartner Jr., Christophe Roeder, Philip V. Ogren, Martha Palmer, and Lawrence Hunter (2010). Annotation of all coreference in biomedical text: Guideline selection and adaptation. In *BioTxtM 2010: 2nd workshop on building and evaluating resources for biomedical text mining*, pages 37–41, Malta.

Cohn, David A., Zoubin Ghahramani, and Michael I. Jordan (1996). Active learning with statistical models. *Journal of Artificial Intelligence Research*, **4**, 129–145.

Converse, Susan (2006). *Pronominal Anaphora Resolution in Chinese*. Ph.D. thesis, Department of Computer and Information Science, University of Pennsylvania, Pittsburgh, Pennsylvania, USA.

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

Dahlmeier, Daniel and Hwee Tou Ng (2010). Domain adaptation for semantic role labeling in the biomedical domain. *Bioinformatics*, **26**(8), 1098–1104.

Daume III, Hal (2006). *Practical Structured Learning for Natural Language Processing*. Ph.D. thesis, University of Southern California, Los Angeles, USA.

Daume III, Hal (2007). Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL2007)*, pages 256–263, Prague, Czech Republic.

Denis, Pascal and Jason Baldridge (2007). Joint determination of anaphoricity and coreference resolution using integer programming. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT2007)*, pages 236–243, Rochester, New York, USA.

Domingos, Pedro (1999). MetaCost: A general method for making classifiers cost-sensitive. In *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD1999)*, pages 155–164, San Diego, California, US.

Donmez, Pinar, Jaime G. Carbonell, and Paul N. Bennett (2007). Dual strategy active learning. In *Proceedings of the 18th European Conference on Machine Learning (ECML2007)*, pages 116–127, Warsaw, Poland.

Elkan, Charles (2001). The foundations of cost-sensitive learning. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI2001)*, Seattle, Washington, USA.

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

Ferrández, Antonio and Jesús Peral (2000). A computational approach to zero-pronouns in Spanish. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL2000)*, pages 166–172, Hong Kong.

Finkel, Jenny Rose and Christopher D. Manning (2008). Enforcing transitivity in coreference resolution. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL2008:HLT), Short Papers*, pages 45–48, Columbus, Ohio, USA.

Finkel, Jenny Rose, Trond Grenager, and Christopher Manning (2005). Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL2005)*, pages 363–370, Ann Arbor, Michigan, USA.

Fisher, David, Stephen Soderland, Joseph McCarthy, Fangfang Feng, and Wendy Lehnert (1995). Description of the UMass system as used for MUC-6. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pages 127–140, Columbia, Maryland, USA.

Florian, Radu, Hany Hassan, Abraham Ittycheriah, Hongyan Jing, Nanda Kambhatla, Xiaoqiang Luo, Nicolas Nicolov, and Salim Roukos (2004). A statistical model for multilingual entity detection and tracking. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL2004)*, pages 1–8, Boston, Massachusetts, USA.

Freund, Yoav and Robert E. Schapire (1999). Large margin classification using the perceptron algorithm. *Machine Learning*, **37**(3), 277–296.

Gasperin, Caroline (2008). *Statistical Anaphora Resolution in Biomedical Texts*. Ph.D. thesis, University of Cambridge, Cambridge, UK.

Gasperin, Caroline (2009). Active learning for anaphora resolution. In *Proceedings of the NAACL-HLT2009 Workshop on Active Learning for Natural Language Processing*, pages 1–8, Boulder, Colorado.

Gasperin, Caroline and Ted Briscoe (2008). Statistical anaphora resolution in biomedical texts. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING2008)*, pages 257–264, Manchester, UK.

Gasperin, Caroline and Renata Vieira (2004). Using word similarity lists for resolving indirect anaphora. In *Proceedings of the ACL2004 Workshop on Reference Resolution and its Applications*, pages 40–46, Barcelona, Spain.

Gasperin, Caroline, Nikiforos Karamanis, and Ruth Seal (2007). Annotation of anaphoric relations in biomedical full-text articles using a domain-relevant scheme. In *Proceedings of the 6th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC2007)*, pages 19–24, Lagos, Portugal.

Ge, Niyu, John Hale, and Eugene Charniak (1998). A statistical approach to anaphora resolution. In *Proceedings of the 6th Workshop on Very Large Corpora (WVLC-6)*, pages 161–170, Montreal, Quebec, Canada.

Grosz, Barbara J., Aravind K. Joshi, and Scott Weinstein (1995). Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, **21**(2), 203–225.

Haghighi, Aria and Dan Klein (2009). Simple coreference resolution with rich syntactic and semantic features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP2009)*, pages 1152–1161, Singapore.

Haghighi, Aria and Dan Klein (2010). Coreference resolution in a modular, entity-centered model. In *Proceedings of the Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT2010)*, pages 385–393, Los Angeles, California.

Halliday, M.A.K. and Ruqaiya Hasan (1976). *Cohesion in English*. Longman Group, London and New York.

Harabagiu, Sanda M., Răzvan C. Bunescu, and Steven J. Maiorano (2001). Text and knowledge mining for coreference resolution. In *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL2001)*, pages 55–62, Pittsburgh, Pennsylvania, USA.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman (2001). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, New York, first edition.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, **57**(1), 97–109.

He, Yuanjian (1998). Zero anaphora, reference tracking and translation. *The Humanities Bulletin*, **5**, 41–47.

Hobbs, Jerry R. (1978). Resolving pronoun references. *Lingua*, **44**, 311–338.

Huang, Shu-Hung (1992). *Zero-Pronouns in Chinese Written Text: Discourse Analysis and Pragmatics*. Ph.D. thesis, Columbia University, New York, USA.

Iida, Ryu, Kentaro Inui, and Yuji Matsumoto (2006). Exploiting syntactic patterns as clues in zero-anaphora resolution. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL2006)*, pages 625–632, Sydney, Australia.

Jiang, Jing and ChengXiang Zhai (2007). Instance weighting for domain adaptation in NLP. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL2007)*, pages 264–271, Prague, Czech Republic.

Joachims, Thorsten (2005). A support vector method for multivariate performance measures. In *Proceedings of the 22nd International Conference on Machine Learning (ICML2005)*, pages 377–384, Bonn, Germany.

Jurafsky, Daniel and James H. Martin (2000). *Speech and Language Processing*. Prentice Hall, New Jersey, USA.

Kawahara, Daisuke and Sadao Kurohashi (2004). Zero pronoun resolution based on automatically constructed case frames and structural preference of antecedents. In *Proceedings of the 1st International Joint Conference on Natural Language Processing (IJCNLP2004)*, pages 12–21, Hainan Island, China.

Kehler, Andrew (1997). Probabilistic coreference in information extraction. In *Proceedings of the the Second Conference on Empirical Methods in Natural Language Processing (EMNLP1997)*, pages 163–173.

Kim, Young-Joo (2000). Subject/object drop in the acquisition of Korean: A cross-linguistic comparison. *Journal of East Asian Linguistics*, **9**(4), 325–351.

Kong, Fang, Guodong Zhou, and Qiaoming Zhu (2009). Employing the centering theory in pronoun resolution from the semantic perspective. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP2009)*, pages 987–996, Singapore.

Lapata, Mirella and Regina Barzilay (2005). Automatic evaluation of text coherence: Models and representations. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI2005)*, pages 1085–1090, Edinburgh, Scotland, UK.

Lee, Cher-Leng (2002). *Zero Anaphor in Chinese*. Crane Publishing, Taipei, Taiwan.

Lewis, David D. and William A. Gale (1994). A sequential algorithm for training text classifiers. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR1994)*, pages 3–12, Dublin, Ireland.

Li, Charles N. and Sandra A. Thompson (1979). Third-person pronouns and zero-anaphora in Chinese discourse. *Syntax and Semantics*, **12**, 311–335.

Li, Wendan (2004). Topic chains in Chinese discourse. *Discourse Processes*, **37**(1), 25–45.

Luo, Xiaoqiang (2005). On coreference resolution performance metrics. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT-EMNLP2005)*, pages 25–32, Vancouver, B.C., Canada.

Marcus, Mitchell P., Beatrice Santorini, and Mary Ann Marcinkiewicz (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, **19**(2), 313–330.

McCarthy, Joseph Francis (1996). *A Trainable Approach to Coreference Resolution for Information Extraction*. Ph.D. thesis, University of Massachusetts Amherst.

McCarthy, Joseph F. and Wendy G. Lehnert (1995). Using decision trees for coreference resolution. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI1995)*, pages 1050–1055, Montréal, Québec, Canada.

Metropolis, Nicholas, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller (1953). Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, **21**(6), 1087–1092.

Morton, Thomas S. (1999). Using coreference to improve passage retrieval for question answering. In *Proceedings of the AAAI1999 Fall Symposium on Question Answering Systems*, pages 72–74.

Moschitti, Alessandro (2006). Making tree kernels practical for natural language learning. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL2006)*, pages 113–120, Trento, Italy.

MUC-6 (1995). Coreference task definition (v2.3, 8 Sep 95). In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pages 335–344, Columbia, Maryland, USA.

MUC-7 (1998). Coreference task definition (v3.0, 13 Jul 97). In *Proceedings of the 7th Message Understanding Conference (MUC-7)*, Fairfax, Virginia, USA.

Na, Seung-Hoon and Hwee Tou Ng (2009). A 2-Poisson model for probabilistic coreference of named entities for improved text retrieval. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR2009)*, pages 275–282, Boston, Massachusetts, USA.

Nakaiwa, Hiromi and Satoru Ikehara (1992). Zero pronoun resolution in a Japanese-to-English machine translation system by using verbal semantic attributes. In *Proceedings of the 3rd Conference on Applied Natural Language Processing (ANLP1992)*, pages 201–208, Trento, Italy.

Nakaiwa, Hiromi and Satoshi Shirai (1996). Anaphora resolution of Japanese zero pronouns with deictic reference. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING1996)*, pages 812–817, Copenhagen, Denmark.

Ng, Vincent (2004a). *Improving Machine Learning Approaches to Noun Phrase Coreference Resolution*. Ph.D. thesis, Cornell University, Ithaca, New York, USA.

Ng, Vincent (2004b). Learning noun phrase anaphoricity to improve coreference resolution: Issues in representation and optimization. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL2004)*, pages 152–159, Barcelona, Spain.

Ng, Vincent (2005). Machine learning for coreference resolution: From local classification to global ranking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL2005)*, pages 157–164, Ann Arbor, Michigan, USA.

Ng, Vincent (2007). Semantic class induction and coreference resolution. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL2007)*, pages 536–543, Prague, Czech Republic.

Ng, Vincent and Claire Cardie (2002a). Combining sample selection and error-driven pruning for machine learning of coreference rules. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP2002)*, pages 55–62, Philadelphia, Pennsylvania, USA.

Ng, Vincent and Claire Cardie (2002b). Identifying anaphoric and non-anaphoric noun phrases to improve coreference resolution. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING2002)*, pages 1–7, Taipei, Taiwan.

Ng, Vincent and Claire Cardie (2002c). Improving machine learning approaches to coreference resolution. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL2002)*, pages 104–111, Philadelphia, Pennsylvania, USA.

NIST (2002). The ACE 2002 evaluation plan. `ftp://jaguar.ncsl.nist.gov/ace/doc/ACE-EvalPlan-2002-v06.pdf`.

Och, Franz Josef (2003). Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL2003)*, pages 160–167, Sapporo, Japan.

Okumura, Manabu and Kouji Tamura (1996). Zero pronoun resolution in Japanese discourse based on centering theory. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING1996)*, pages 871–876, Copenhagen, Denmark.

Poesio, Massimo, Olga Uryupina, Renata Vieira, Mijail Alexandrov-Kabadjov, and Rodrigo Goulart (2004). Discourse-new detectors for definite description resolution: A survey and a preliminary proposal. In *Proceedings of the ACL2004 Workshop on Reference Resolution and its Applications*, pages 47–54, Barcelona, Spain.

Ponzetto, Simone Paolo and Michael Strube (2006). Exploiting semantic role labeling, WordNet and Wikipedia for coreference resolution. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL2006)*, pages 192–199, New York City, USA.

Quinlan, J. Ross (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Francisco, California, USA.

Rai, Piyush, Avishek Saha, Hal Daume, and Suresh Venkatasubramanian (2010). Domain adaptation meets active learning. In *Proceedings of the NAACL-HLT2010 Workshop on Active Learning for Natural Language Processing*, pages 27–32, Los Angeles, California.

Russell, Stuart and Peter Norvig (2002). *Artificial Intelligence: A Modern Approach*. Prentice Hall, New Jersey, second edition.

Seki, Kazuhiro, Atsushi Fujii, and Tetsuya Ishikawa (2002). A probabilistic method for analyzing Japanese anaphora integrating zero pronoun detection and resolution. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING2002)*, pages 911–917, Taipei, Taiwan.

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

Steinberger, Josef, Mijail Kabadjov, Massimo Poesio, and Olivia Sanchez-Graillet (2005). Improving LSA-based summarization with anaphora resolution. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT-EMNLP2005)*, pages 1–8, Vancouver, B.C., Canada.

Stoyanov, Veselin and Claire Cardie (2006). Partially supervised coreference resolution for opinion summarization through structured rule learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP2006)*, pages 336–344, Sydney, Australia.

Stoyanov, Veselin, Nathan Gilbert, Claire Cardie, and Ellen Riloff (2009). Conundrums in noun phrase coreference resolution: Making sense of the state-of-the-art. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing(ACL-IJCNLP2009)*, pages 656–664, Singapore.

Tang, Min, Xiaoqiang Luo, and Salim Roukos (2002). Active learning for statistical natural language parsing. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL2002)*, pages 120–127, Philadelphia, Pennsylvania, USA.

Tao, Liang and Alice F. Healy (2005). Zero anaphora: Transfer of reference tracking strategies from Chinese to English. *Journal of Psycholinguistic Research*, **34**(2), 99–131.

van Deemter, Kees and Rodger Kibble (2000). On coreferring: Coreference in MUC and related annotation schemes. *Computational Linguistics*, **26**(4), 629–637.

Vemulapalli, Smita, Xiaoqiang Luo, John F. Pitrelli, and Imed Zitouni (2009). Classifier combination techniques applied to coreference resolution. In *Proceedings of theHuman Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT2009), Companion Volume: Student Research Workshop and Doctoral Consortium*, pages 1–6, Boulder, Colorado.

Versley, Yannick, Simone Paolo Ponzetto, Massimo Poesio, Vladimir Eidelman, Alan Jern, Jason Smith, Xiaofeng Yang, and Alessandro Moschitti (2008a). BART: A modular

toolkit for coreference resolution. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL2008:HLT), Demo Session*, pages 9–12, Columbus, Ohio, USA.

Versley, Yannick, Simone Paolo Ponzetto, Massimo Poesio, Vladimir Eidelman, Alan Jern, Jason Smith, Xiaofeng Yang, and Alessandro Moschitti (2008b). BART: A modular toolkit for coreference resolution. In *Proceedings of the the Sixth International Language Resources and Evaluation (LREC2008)*, pages 962–965, Marrakech, Morocco.

Vieira, R., E. Bick, J. Coelho, V. Muller, S. Collovini, J. Souza, and L. Rino (2006). Semantic tagging for resolution of indirect anaphora. In *Proceedings of the 7th SIGdial Workshop on Discourse and Dialogue*, pages 76–79, Sydney, Australia.

Vilain, Marc, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman (1995). A model-theoretic coreference scoring scheme. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pages 45–52, Columbia, Maryland, USA.

Wang, Chi-shing and Grace Ngai (2006). A clustering approach for unsupervised Chinese coreference resolution. In *Proceedings of the 5th SIGHAN Workshop on Chinese Language Processing*, pages 40–47, Sydney, Australia.

Wasserman, Larry (2004). *All of Statistics: A Concise Course in Statistical Inference*. Springer Texts in Statistics. Springer, New York, first edition.

Wick, Michael and Andrew McCallum (2009). Advances in learning and inference for partition-wise models of coreference resolution. Technical Report UM-CS-2009-028, University of Massachusets, Amherst, Massachusetts, USA.

Witte, René and Sabine Bergler (2003). Fuzzy coreference resolution for summarization.

In *Proceedings of the International Symposium on Reference Resolution and Its Applications to Question Answering and Summarization (ARQAS 2003)*, pages 43–50, Venice, Italy.

Witten, Ian H. and Eibe Frank (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. The Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann Publishers, San Francisco, California, USA, second edition.

Xue, Nianwen, Fei Xia, Fu-Dong Chiou, and Martha Palmer (2005). The Penn Chinese TreeBank: Phrase structure annotation of a large corpus. *Natural Language Engineering*, **11**(2), 207–238.

Yang, Xiaofeng, Guodong Zhou, Jian Su, and Chew Lim Tan (2003). Coreference resolution using competition learning approach. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL2003)*, pages 176–183, Sapporo, Japan.

Yang, Xiaofeng, Guodong Zhou, Jian Su, and Chew Lim Tan (2004a). Improving noun phrase coreference resolution by matching strings. In *Proceedings of the 1st International Joint Conference on Natural Language Processing (IJCNLP2004)*, pages 22–31, Hainan Island, China.

Yang, Xiaofeng, Jian Su, Guodong Zhou, and Chew Lim Tan (2004b). Improving pronoun resolution by incorporating coreferential information of candidates. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL2004)*, pages 127–134, Barcelona, Spain.

Yang, Xiaofeng, Jian Su, Guodong Zhou, and Chew Lim Tan (2004c). An NP-cluster based

approach to coreference resolution. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING2004)*, pages 226–232, Geneva, Switzerland.

Yeh, Ching-Long and Yi-Chun Chen (2004). Zero anaphora resolution in Chinese with shallow parsing. *Journal of Chinese Language and Computing*.

Zadrozny, Bianca and Charles Elkan (2001). Learning and making decisions when costs and probabilities are both unknown. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD2001)*, pages 204–213, San Francisco, California, USA.

Zadrozny, Bianca, John Langford, and Naoki Abe (2003). Cost-sensitive learning by cost-proportionate example weighting. In *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM2003)*, pages 435–442, Melbourne, Florida, USA.

Zelenko, Dmitry, Chinatsu Aone, and Jason Tibbetts (2004). Coreference resolution for information extraction. In *Proceedings of the ACL2004 Workshop on Reference Resolution and its Applications*, pages 24–31, Barcelona, Spain.

Zhao, Shanheng and Hwee Tou Ng (2007). Identification and resolution of Chinese zero pronouns: A machine learning approach. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL2007)*, pages 208–215, Prague, Czech Republic.

Zhao, Shanheng and Hwee Tou Ng (2010). Maximum metric score training for coreference resolution. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING2010)*, pages 1308–1316, Beijing, China.

Zhong, Zhi, Hwee Tou Ng, and Yee Seng Chan (2008). Word sense disambiguation using OntoNotes: An empirical study. In *Proceedings of the 2008 Conference on Empirical*

Methods in Natural Language Processing (EMNLP2008), pages 1002–1010, Honolulu, Hawaii, USA.

Zhou, Guodong, Jie Zhang, Jian Su, Dan Shen, and Chew Lim Tan (2004). Recognizing names in biomedical texts: A machine learning approach. *Bioinformatics*, **20**(7), 1178–1190.

Zhou, Yaqian, Changning Huang, Jianfeng Gao, and Lide Wu (2005). Transformation based Chinese entity detection and tracking. In *Proceedings of the 2nd International Joint Conference on Natural Language Processing (IJCNLP2005)*, pages 232–237, Jeju Island, Korea.

Zhu, Jingbo, Huizhen Wang, Benjamin K. Tsou, and Matthew Ma (2010). Active learning with sampling by uncertainty and density for data annotations. *IEEE Transactions on Audio, Speech, and Language Processing*, **18**(6), 1323–1331.