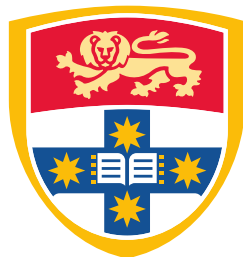


# Improved Coreference Resolution Using Cognitive Insights

Kellie Webster



THE UNIVERSITY OF  
**SYDNEY**

A thesis submitted in fulfilment of the requirements for the degree of Doctor of Philosophy

The University of Sydney

School of Information Technologies

2016



# Abstract

Coreference resolution is the task of extracting referential expressions, or mentions, in text and clustering these by the entity or concept they refer to. It is an important component of natural language processing (NLP) pipelines since it enables computational systems to understand that the information from textual attributes and relationships of mentions concern coherent entities, such as particular people or organisations. The sustained research interest in the task reflects the richness of reference expression usage in natural language and the difficulty in encoding insights from linguistic and cognitive theories effectively.

In this thesis, we design and implement **LIMERIC**, a state-of-the-art coreference resolution engine. In the literature, coreference resolution has typically been modelled as a mention pairing problem. However, simple local decoding strategies make errors by failing to account for global consistency constraints, and the two directions to incorporate such constraints – non-local decoding and entity-level modelling – have largely been orthogonal. **LIMERIC** naturally incorporates both to achieve the highly competitive benchmark performance of 64.22% and 59.99% using gold and automatic preprocessing on the CoNLL-2012 benchmark with a simple model and a baseline feature set. This performance is stronger than any system that only use non-local decoding or entity-level modelling in isolation for global consistency, arguing for their mutual benefit.

As well as strong performance, a key contribution of this work is a reconceptualisation of the coreference task. We draw an analogy between shift-reduce parsing and coreference resolution to develop an algorithm which naturally mimics cognitive models of human discourse processing. Leveraging the self-ordering forest of discourse entities as a simple model of the human mind, we redefine how features can be defined and competition in antecedent selection modelled.

In our feature development work, we leverage insights from cognitive theories to improve our modelling. Specifically, we exploit the fine-grained typology of the Acces-

sibility hierarchy (Ariel, 2001), as well as a range of factors postulated to explain human reference resolution: antecedent competition, frame semantic inference, and syntactic parallelism. Each contribution achieves statistically significant improvements and sum to gains of 1.65% and 1.66% on the CoNLL-2012 benchmark, yielding performance values of 65.76% and 61.27%. This performance is either better or not significantly different from our benchmark, Björkelund and Kuhn (2014), the best performing system at the time of this work.

For each novel feature we propose, we contribute an accompanying analysis so as to better understand how cognitive theories apply to real language data. These enable us to identify fine-grained patterns in reference expression usage, to demonstrate the insufficiency of cohesion for modelling coreference, and to identify factors contributing to the difficulty in achieving performance gains from using frame semantic knowledge.

The techniques we propose in this thesis represent a break from how coreference resolution has been approached as a computational task; LIMERIC is at once a platform for exploring cognitive insights into coreference and a viable alternative to current systems. We are excited by the promise of incorporating our and further cognitive insights into more complex frameworks since this has the potential to both improve the performance of computational models, as well as our understanding of the mechanisms underpinning human reference resolution. By furthering our understanding of how to model coreference, we improve our ability to organise and leverage the huge amounts of information expressed in collections of natural language data.

# Acknowledgements

After an eleven year career as an ‘eternal student’, there are many people to thank.

First, I would like to thank the Schwa Lab who helped me add the adjective in ‘computational linguist’ over these past three-and-a-bit years. While every member, past and present, shaped my progress in some way, special thanks go to James Curran, for introducing me to programming in his fabulous Informatics course and (slowly) persuading me of my potential in the field; to Will Radford and Dominick Ng, first my tutors in the course and always invaluable gurus; to Joel Nothman for his commitment to academic rigour and many stimulating discussions; to Nicky Ringland for being my thesis writing buddy, through thick and thin; and, finally, to Glen Pink and Ben Hachey for making my experience so enjoyable, for many coffees and far too many hours of annotation and proof-reading.

My work has benefited from interaction with the wider research and industry community. I thank the Girls Programming Network and the Anita Borg and Grace Hopper communities for introducing me to an inspiring group of technical women. I also am grateful to everyone in the executive and broader community of the Australasian Language Technology Association, and my colleagues at Google and Hugo; the fabulous experience of working with you all has truly made deciding between potential next steps particularly difficult. The School of Information Technologies at the University of Sydney has provided me with funding to attend conferences and access to a thriving research community, in which Alan Fekete stands out as deserving thanks for always being available for advice.

The trials of a PhD can only be borne when balanced by incredible friends and family. I name here Kristina Gatt, Maxine Roff, Michelle Sauvignano, Mahboobeh Moghaddam and Kittiya Muaksang and families, Rebekah Wegener, Tamsin Peters, Emily O'Brien, and Elizabeth Leung. Despite my being at times pre-occupied with seemingly minute details, all of you have been invaluable sources of laughter, stress relief, distraction, wine, and tea. At home, I thank my Mum for taking a poor student back into an empty nest and everything this entailed, my sister Erin for great company over Sunday dinners and the weekend quiz, and my brother and sister-in-law, Danny and Mi Ra. I could not have done this without you.

# Statement of compliance

I certify that:

- I have read and understood the University of Sydney Student Plagiarism: Coursework Policy and Procedure;
- I understand that failure to comply with the Student Plagiarism: Coursework Policy and Procedure can lead to the University commencing proceedings against me for potential student misconduct under Chapter 8 of the University of Sydney By-Law 1999 (as amended);
- this Work is substantially my own, and to the extent that any part of this Work is not my own I have indicated that it is not my own by Acknowledging the Source of that part or those parts of the Work.

Name: *Kellie Webster*

Signature:

Date: *8th January, 2016*



# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Challenges in Coreference Resolution . . . . .	4
1.1.1	Modelling . . . . .	4
1.1.2	Cognitive Insights . . . . .	5
1.2	Contributions . . . . .	6
<b>2</b>	<b>Task Definition</b>	<b>9</b>
2.1	Standard Datasets . . . . .	9
2.1.1	Message Understanding Conferences . . . . .	10
2.1.2	Automatic Content Extraction . . . . .	14
2.1.3	OntoNotes . . . . .	17
2.2	Evaluation . . . . .	21
2.2.1	MUC . . . . .	22
2.2.2	B <sup>3</sup> . . . . .	24
2.2.3	CEAF . . . . .	26
2.2.4	BLANC . . . . .	28
2.2.5	CoNLL . . . . .	30
2.2.6	Error-Driven Evaluation . . . . .	30
2.3	Summary . . . . .	32
<b>3</b>	<b>Background</b>	<b>33</b>
3.1	Models . . . . .	33

3.1.1	Linguistic Reference . . . . .	34
3.1.2	Rule-Based Discourse Models . . . . .	37
3.1.3	Mention-Pair Models . . . . .	38
3.1.4	Entity-Level Models . . . . .	45
3.1.5	Structured Prediction . . . . .	51
3.1.6	Summary . . . . .	53
3.2	Features . . . . .	54
3.2.1	Linguistic Description . . . . .	55
3.2.2	Feature Review . . . . .	63
3.2.3	Summary . . . . .	75
3.3	Summary . . . . .	76
<b>4</b>	<b>Incremental Coreference Resolution</b>	<b>77</b>
4.1	Motivation . . . . .	78
4.1.1	Shift-Reduce Parsing . . . . .	78
4.1.2	Cognitive Insight . . . . .	81
4.1.3	Entity-Centric Design . . . . .	82
4.1.4	Anaphoricity Determination . . . . .	83
4.2	System Design . . . . .	85
4.2.1	Initialisation . . . . .	85
4.2.2	Inference . . . . .	88
4.2.3	Training . . . . .	91
4.3	Features . . . . .	96
4.3.1	Implementation . . . . .	96
4.3.2	Analysis . . . . .	103
4.4	Evaluation . . . . .	108
4.4.1	Benchmarking . . . . .	108
4.4.2	Error Analysis . . . . .	110
4.5	Summary . . . . .	112

<b>5</b>	<b>Accessibility Hierarchy</b>	<b>113</b>
5.1	Research Questions . . . . .	114
5.2	Experimental Setup . . . . .	115
5.2.1	Mention Classification . . . . .	115
5.2.2	Discourse Transition Pairs . . . . .	118
5.3	Trends in OntoNotes 5.0 . . . . .	118
5.3.1	Discourse Transition Trends . . . . .	119
5.3.2	Anaphoricity Trends . . . . .	122
5.4	Evaluation . . . . .	124
5.4.1	Feature Prefixes . . . . .	124
5.4.2	Benchmarking . . . . .	131
5.4.3	Error Analysis . . . . .	132
5.5	Summary . . . . .	133
<b>6</b>	<b>Mutually Informative Features</b>	<b>135</b>
6.1	Motivation . . . . .	136
6.1.1	Antecedent Competition . . . . .	136
6.1.2	Feature Non-Independence . . . . .	137
6.2	Secondary Features . . . . .	138
6.2.1	Association Statistics . . . . .	138
6.2.2	Observed Associations . . . . .	141
6.2.3	Secondary Features . . . . .	149
6.3	Feature Competition . . . . .	155
6.3.1	Experimental Setup . . . . .	156
6.3.2	Forms of Competition . . . . .	156
6.4	Evaluation . . . . .	162
6.4.1	Benchmarking . . . . .	162
6.4.2	Error Analysis . . . . .	164
6.5	Summary . . . . .	165

<b>7</b>	<b>Frame Semantic Inference</b>	<b>167</b>
7.1	Background . . . . .	168
7.1.1	Winograd Schema Challenge . . . . .	168
7.1.2	Frame Semantic Resources . . . . .	169
7.1.3	Brown Clustering . . . . .	172
7.2	Frame Semantic Resources . . . . .	173
7.2.1	Predicate Clustering . . . . .	173
7.2.2	Argument Selection . . . . .	177
7.2.3	Inter-Resource Comparison . . . . .	179
7.2.4	Summary . . . . .	180
7.3	Feature Development . . . . .	181
7.3.1	Feature Variants . . . . .	181
7.3.2	FrameNet . . . . .	182
7.3.3	Narrative Schema . . . . .	184
7.3.4	Brown Clusters . . . . .	186
7.3.5	All Resources . . . . .	190
7.4	Evaluation . . . . .	192
7.4.1	Benchmarking . . . . .	192
7.4.2	Error Analysis . . . . .	193
7.5	Summary . . . . .	195
<b>8</b>	<b>Conclusion</b>	<b>197</b>
8.1	Future Work . . . . .	198
8.1.1	Robust Models of Coreference . . . . .	198
8.1.2	Insufficiency of Cohesion . . . . .	199
8.1.3	Extending Frame Semantic Inference . . . . .	200
8.1.4	Further Insights from Psycholinguistic Theories . . . . .	201
8.1.5	Languages Other Than English . . . . .	201
8.2	Summary . . . . .	202





# List of Figures

2.1	Example output for evaluation. . . . .	21
2.2	Example output for BLANC evaluation. . . . .	28
3.1	Example of c-command relationship in a constituency parse structure. .	56
3.2	Centering analysis of the example excerpt. . . . .	59
3.3	Accessibility hierarchy of Ariel (2001). . . . .	61
4.1	Series of shift and reduce operations creating a syntactic parse tree. . .	79
4.2	Series of shift and reduce operations creating a collection of emerging entity clusters. . . . .	81
4.3	Determining the correct classification from gold standard annotations. .	94
4.4	Number of distinct features and their average weight in LIMERIC, by feature class and subclass. . . . .	105
4.5	Depth in forest of correct prediction in CoNLL-2012 DEV using gold preprocessing. . . . .	107
4.6	Errors made by LIMERIC and the current state of the art, IMS, on CoNLL- 2012 TEST using gold preprocessing. . . . .	111
4.7	Errors made by LIMERIC and the current state of the art, IMS, on CoNLL- 2012 TEST using automatic preprocessing. . . . .	111
5.1	Accessibility hierarchy of Ariel (2001). . . . .	114
5.2	Errors made by AR Transitions model compared to our LIMERIC base- line and IMS on CoNLL-2012 TEST using gold preprocessing. . . . .	132

5.3	Errors made by <i>AR</i> Transitions model compared to our LIMERIC base-line and IMS on CoNLL-2012 TEST using automatic preprocessing. . . .	133
6.1	Example of stack competition feature extraction. . . . .	158
6.2	Errors made by our Mutual Information model compared to our previous models and IMS on CoNLL-2012 TEST using gold preprocessing. . . .	164
6.3	Errors made by our Mutual Information model compared to our previous models and IMS on CoNLL-2012 TEST using automatic preprocessing. .	165
7.1	Errors made by Same 8-Prefix model compared to our previous baselines on CoNLL-2012 TEST using gold preprocessing. . . . .	194
7.2	Errors made by Same 8-Prefix model compared to our previous baselines on CoNLL-2012 TEST using automatic preprocessing. . . . .	195

# List of Tables

2.1	Overview of the differences between MUC, ACE, and OntoNotes coreference annotations. . . . .	10
2.2	Coreference annotation statistics for the MUC corpora. . . . .	11
2.3	Coreference annotation statistics for the (English) ACE corpora. The number of words is as reported by Doddington et al. (2004) and NIST (2005). . . . .	15
2.4	Coreference annotation statistics for (English) OntoNotes 5. . . . .	18
2.5	Official scores of the competing systems at CoNLL-2012. . . . .	31
3.1	Performance of rule-based entity level models on MUC-7. . . . .	38
3.2	Performance of mention-pair models on standard evaluation corpora. . . . .	39
3.3	Performance of entity-level models on standard evaluation corpora. * indicates results are ACE scores using gold mentions. . . . .	46
3.4	Performance of structured prediction models on standard evaluation corpora. . . . .	51
4.1	Number of mentions extracted from the TRAIN and DEV portions of OntoNotes 5, using two-stage mention extraction. . . . .	87
4.2	Number of mentions extracted from the TRAIN and DEV portions of OntoNotes 5, using three-stage mention extraction. . . . .	88
4.3	Baseline feature set of LIMERIC. . . . .	97
4.4	Ablation analysis over CoNLL-2012 DEV using gold preprocessing. . . . .	103

4.5	Performance of LIMERIC on CoNLL-2012 TEST. . . . .	109
5.1	Accessibility rank values used in our experiments, with their base distribution over extracted NPs. . . . .	116
5.2	Most common mention strings for each accessibility rank value. . . . .	116
5.3	Accessibility transitions in CoNLL-2012 DEV by accessibility rank value.	120
5.4	Proportion of singletons in CoNLL-2012 DEV by accessibility rank value.	123
5.5	Performance of AR feature prefixing on CoNLL-2012 DEV. . . . .	124
5.6	Number of distinct features and their average weight in our AR Transitions model, compared to LIMERIC. . . . .	126
5.7	Ten most positively weighted features in our AR Transitions model. . .	128
5.8	Ten most negatively weighted features in our AR Transitions model. . .	129
5.9	Number of distinct features and their average weight in our AR Rankings model, compared to LIMERIC. . . . .	130
5.10	Performance of AR Transition prefixing on CoNLL-2012 TEST. . . . .	131
6.1	Matrix of outcomes over two possible feature extractions. . . . .	139
6.2	$\chi^2$ values for different pairings of head match and head POS tag features.	142
6.3	Proportion of head matched nominal mention pairs which are coreferential.	143
6.4	$\chi^2$ statistics for different pairings of surface form cohesion and proximity features. . . . .	144
6.5	$\chi^2$ statistics for pairs of attribute match features. . . . .	146
6.6	$\chi^2$ statistics for different pairings of attribute cohesion and proximity features. . . . .	147
6.7	$\chi^2$ statistics for the association of topicality and our various cohesion features. . . . .	148
6.8	Secondary feature set of conjunctive features. . . . .	149
6.9	Performance of secondary features on CoNLL-2012 DEV. . . . .	150

6.10	Weights of unprefixed surface form cohesion and depth secondary features in the Surface + Depth model. . . . .	151
6.11	Weights of unprefixed features conjoining document genre and head word match, in the Head Match model. . . . .	152
6.12	Weights of the unprefixed paired attribute match features on Attribute Pairs model. . . . .	154
6.13	Weight of the unprefixed features conjoining attribute match with cluster length in our Attribute + Topicality model. . . . .	155
6.14	Performance of competition features on CoNLL-2012 DEV. . . . .	159
6.15	Average weight of anaphoricity competition features. . . . .	160
6.16	Performance of secondary (AR Transitions) and competition (Mutual Information) features on CoNLL-2012 DEV. . . . .	162
6.17	Performance of secondary and competition features on CoNLL-2012 TEST.	163
7.1	Coverage of mentions by the proposed frame semantics resources. . . .	174
7.2	Coverage of mention-pair links by the proposed frame semantics resources.	176
7.3	Number of subject and object mentions in pairs related by the proposed frame semantic resources. . . . .	178
7.4	Overlap between proposed frame semantic resources. . . . .	179
7.5	Performance of FrameNet features on CoNLL-2012 DEV with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information). . . . .	183
7.6	Performance of Narrative Schema features on CoNLL-2012 DEV with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information). . . . .	184
7.7	Performance of Brown cluster features (using sparse representation over 3200 cluster data) on CoNLL-2012 DEV with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information). . . . .	187

7.8	Performance of Brown cluster features (using sparse representation over length 12 prefixes) on CoNLL-2012 DEV with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information). . . . .	188
7.9	Performance of Brown cluster features (using length 12 prefixes over 3200 cluster data) on CoNLL-2012 DEV with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information). . . . .	190
7.10	Performance of combined model using all frame semantic resource features on CoNLL-2012 DEV with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information). . . . .	191
7.11	Performance of Brown cluster features on CoNLL-2012 TEST with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information). . . . .	192



# 1 Introduction

Natural language processing (NLP) is concerned with building automatic systems which are able to understand natural language data. Its goal of human-level language comprehension is particularly attractive given the large volume of natural language data available, for instance via the internet; automatic systems which are able to intelligently process this data would enable us to extract and organise the vast amount of information it expresses. The high-level goal of language comprehension is typically decomposed into a number of sub-tasks which may be composed into a pipeline solution. The focus of this thesis is a core sub-task common to many NLP pipelines, coreference resolution.

Coreference resolution is the task of extracting referential expressions, or *mentions*, in text and clustering these according to the entity or concept they refer to. For instance, in the following Voice of America excerpt, an ideal coreference resolution engine would produce a cluster containing the mentions *'The battered US Navy destroyer Cole'*, *'its'*, and *'the ship'* and this cluster would be distinct from the one containing *'a huge Norwegian transport vessel'*, which refers to a related, but distinct, entity.

**The battered US Navy destroyer Cole** has begun **its** journey home from Yemen, 17 days after a suspected terrorist bomb tore a hole in **its** side. The attack killed 17 American soldiers and wounded 39. Flanked by other US warships and guarded by aircraft, **the ship** was towed out of Aden Harbor to rendezvous with *a huge Norwegian transport vessel* that will carry **the crippled ship** to the United States.

## 1.1 Challenges in Coreference Resolution

Coreference resolution is an active area of research. This interest reflects the challenges involved in developing computational systems to effectively capture the richness of reference expression usage in natural language.

### 1.1.1 Modelling

Coreference resolution has typically been approached using the *mention-pair* model in which each pairing of extracted mentions is evaluated for their compatibility based on defined linguistic indicators. These  $O(n^2)$  scores then need to be decoded into a clustering over mentions; the complete enumeration of possible clusterings of mentions is exponentially large and this has motivated the use of greedy algorithms.

A simple decoding strategy is to greedily cluster compatible mentions which are close to one another in their source document. This strategy serves to establish a reasonable baseline since textual proximity is indeed an indicator of coreference, but can make globally inconsistent decisions. For instance, ‘*The battered US Navy destroyer Cole*’ and ‘*the ship*’ may be highly compatible, but so too may be ‘*the ship*’ and ‘*a huge Norwegian transport vessel*’; if these resolutions are done independently of one another, we may erroneously corefer ‘*The battered US Navy destroyer Cole*’ and ‘*a huge Norwegian transport vessel*’.

Two promising but orthogonal approaches to incorporate global consistency into coreference modelling are *mention synchronous* or *non-local* decoding (Ng and Cardie, 2002b; Durrett and Klein, 2013; Chang et al., 2013) and *entity-level* modelling (Rahman and Ng, 2009; Raghunathan et al., 2010). Non-local decoding refers to strategies which cluster mentions based on overall compatibility, rather than just textual proximity; entity-level modelling refers to algorithms which incrementally grow entity clusters, which allows feature extraction to be aware of previous resolutions. Both of these approaches could improve our resolution of the example excerpt. For instance, ‘*the*

*crippled ship*' and the *'the ship'* refer to the same entity using similar words despite being separated textually by the distractor mention *'a huge Norwegian transport vessel'*; this similarity would allow a non-local decoding strategy to prefer this resolution. In a similar vein, if *'the ship'* has already been resolved to *'The battered US Navy destroyer Cole'*, nationality modification argues against *'a huge Norwegian transport vessel'* also joining this cluster. Among current systems, decoding strategies are increasingly complex and entity-level models do not fully leverage psycholinguistic cues such as reading order. Structured prediction offers a means to incorporate both, but is rigid in how entity-level features may be defined.

### 1.1.2 Cognitive Insights

Coreference resolution may be defined at the level of the document or across a collection of documents; this thesis is concerned with the former. The clusters produced therefore correspond to the *discourse* entities around which the narrative of the document develops. In psycholinguistic theory, discourse entities are distinct from real-world entities in that they are abstract and have properties that are incrementally developed as a discourse proceeds. That is, coreference is a relationship between a mention and a grouping of entity mentions from the proceeding discourse, and its resolution follows the natural top-to-bottom, left-to-right reading order of documents. While outside the scope of our work, we note that discourse entities, particularly those headed by a proper name, are anchored in the real world, a fact which motivates joint models of Named Entity Linking and coreference resolution (e.g. Hajishirzi et al., 2013).

Having discourse entities as our object of interest allows us to draw insights from psycholinguistic and cognitive theories including Centering (Grosz et al., 1995) and Accessibility (Ariel, 2001). The richness in these theories, including Accessibility theory's hierarchy of reference expressions and the explanatory factors of cohesion, proximity, parallelism, topicality, competition, and inference automaticity, have yet to be fully

explored for their utility in computational modelling. It remains an open question whether these insights can improve the performance of our computational systems.

## 1.2 Contributions

The main contribution of this thesis is a reconceptualisation of computational approaches to coreference resolution. We draw an analogy between coreference resolution and shift-reduce parsing to develop an incremental clustering algorithm which is able to leverage the strengths of both non-local decoding and entity-level modelling for global consistency. As well as yielding an efficient and simple model, our baseline system is highly competitive with the current state of the art. Furthermore, the forest of discourse entities in this model can be viewed as a simple model of the human mind, giving us the opportunity to explore defining features and modelling the competition between candidate antecedents in a cognitively-aware way.

Concretely, we design and build a coreference resolution engine, *LIMERIC*, in Chapter 4. *LIMERIC*'s baseline configuration achieves 64.22% and 59.99% on the standard CoNLL-2012 benchmark, using gold and automatic preprocessing. This performance is competitive with the best reported research systems and outperforms all systems which use just non-local decoding or entity-level modelling to capture global consistency, arguing for their mutual benefit.

Chapters 5, 6, and 7 improve from this strong baseline by exploiting insights from cognitive theory. Chapter 5 incorporates the fine-grained mention hierarchy of Accessibility theory (Ariel, 2001); Chapter 6 considers the mutual information in features, which includes how antecedent competition can be modelled in a cognitively-aware way; and Chapter 7 adapts features from the Winograd Schema Challenge to capture frame semantic inference in a natural discourse setting.

Improvements in Chapter 5 from incorporating the fine-grained Accessibility hierarchy yield a statistically significant improvement on both gold and automatic settings

of CoNLL-2012 against LIMERIC, while those in Chapter 6 from mutual information are additionally significant above this enriched baseline on the difficult automatic setting. Our best performing system from these closed-task chapters achieves CoNLL-2012 scores of 65.29% and 61.13% using gold and automatic preprocessing. This performance is either better or not significantly different from Björkelund and Kuhn (2014), the best reported system performance on the benchmark task at the time of this work<sup>1</sup>. Furthermore, the analyses accompanying these feature proposals contributes to our understanding of the fine-grained trends in reference expression usage, as well as the complex interactions of coreference indicators. This work is valuable for understanding the mechanisms underpinning human reference resolution, which in turn sheds light on how to improve computational systems for the task. Specifically, we argue that the degree of cohesion between mentions is insufficient for resolving reference and provide detailed analyses of the utility of a wider set of cognitively-aware indicators.

Error analysis reveals that LIMERIC makes errors from being overly conservative. We identify frame semantic inference as a promising way to address this and explore its challenges in Chapter 7. We find that the two commonly used frame semantic resources, FrameNet and Narrative Schemas, suffer from poor coverage, and propose Brown clusters as an automatically generated alternative to these. Despite being simple to extract, Brown cluster features outperform those based on FrameNet and Narrative Schema, though we fail to find mutual benefit from using multiple resources. This work achieves a weakly significant improvement on the gold setting of the CoNLL-2012 benchmark and opens up the possibility of exploring frame semantic inference in under-resourced settings. We see future work in expanding resources and modelling their non-independent views on frame semantic knowledge.

---

<sup>1</sup>As noted in Chapter 3, the current best reported performance is Wiseman et al. (2015)



## 2 Task Definition

The formulation of coreference resolution as a computational problem has largely been shaped by its definition in shared tasks. While the current standard, which is used in this thesis, is the definition of OntoNotes, evaluated at Conferences on Natural Language Learning (CoNLL), it is important to understand how this definition has developed from those used earlier, for the Message Understanding Conference (MUC) and Automatic Content Extraction (ACE) projects, since issues raised in these efforts have introduced changes which have implications for system design.

Section 2.1 reviews the development of the task guidelines from the perspective of the annotation guidelines and the datasets labelled with these. Referential ambiguity is highlighted as a persistent problem for annotation. While the shared task format and evaluation gives us a stable basis for comparison, all proposed coreference resolution metrics to date have observed biases. The metrics available and a discussion of their biases forms Section 2.2, which concludes with a discussion of remaining challenges. In particular, we review the error-driven evaluation method proposed by Kummerfeld and Klein (2013) which is designed to give a finer-grained analysis of system output, with the aim of informing research design.

### 2.1 Standard Datasets

The development of coreference resolution as a computational task has been shaped by three shared tasks at conferences targeting information extraction. Each has associated datasets on which systems may be developed and evaluated, summarised in Table 2.1.

Task	Agreement	Dataset	
		Languages	Genres
MUC	91%	1	1
ACE	86%	3	3
OntoNotes	88-94%	3	7

Table 2.1: Overview of the differences between MUC, ACE, and OntoNotes coreference annotations.

We can see that the trend is for datasets to grow in size and scope, while guidelines are refined to maintain reasonable agreement between annotators (measured using F-score). We review each in turn, chronologically. English is a target for all these efforts and the focus of this work. Therefore, we do not review non-English corpora here.

### 2.1.1 Message Understanding Conferences

Coreference resolution was first formulated as a shared task in 1995 at the 6th Message Understanding Conference (MUC-6; Grishman and Sundheim, 1996), where it complemented the conference’s core task of template filling, in which systems produced structured information stores, or templates, about people and organisations.

Coreference resolution was introduced to address one of the three key goals of MUC-6, namely to encourage work on “deeper understanding” of documents. Coordinators saw the reliance at previous conferences on local pattern matching for template filling as problematic; the introduction of coreference resolution was intended to promote semantically richer modelling of documents.

**Dataset** The MUC corpora are derived from newswire, primarily the Wall Street Journal for MUC-6<sup>1</sup>, with additional material from Reuters<sup>2</sup>, and the New York Times for MUC-7<sup>3</sup>. Statistics over the two are given in Table 2.2; columns represent the number

<sup>1</sup><https://catalog.ldc.upenn.edu/LDC2003T13>

<sup>2</sup><https://catalog.ldc.upenn.edu/LDC96T10>

<sup>3</sup><https://catalog.ldc.upenn.edu/LDC2001T02>

Dataset	Train	Test	Tokens	Mentions	Clusters	M/C
MUC-6	30	30	27,059	4,232	960	4.4
MUC-7	30	20	27,996	4,297	1,081	3.9

Table 2.2: Coreference annotation statistics for the MUC corpora.

of documents in the training and test splits, the number of mentions and entity clusters in the combined dataset, and the mean number of mentions per entity cluster.

Compared to later datasets, the MUC datasets are small and this limits their usefulness for modern approaches to coreference resolution, especially learning-based approaches. They are still used, though infrequently, to benchmark performance against seminal work evaluated at, and following, MUC.

**Annotation** Annotation was carried out by a team of experts in computational linguistics with the goal of clarifying how to define the task and identifying problems in annotating coreference relationships (6th Message Understanding Conference, 1995; 7th Message Understanding Conference, 1997). While the initial goal of the annotation was to mark three coreference relationships, namely identity of reference, part-whole, and set-subset, only identity ended up being annotated. This was because the annotation task, as well as the task of devising consistent task guidelines, was found to be more difficult than anticipated (Grishman and Sundheim, 1996).

**The battered US Navy destroyer Cole** has begun **its** journey home from Yemen, 17 days after a suspected terrorist bomb tore a hole in **its** side. The attack killed 17 American soldiers and wounded 39. Flanked by other US warships and guarded by aircraft, **the ship** was towed out of Aden Harbor to rendezvous with a huge Norwegian transport vessel that will carry **the crippled ship** to the United States.

Coreference was modelled as a pairwise relationship between two nouns. For instance, since ‘*The US Navy destroyer Cole*’ and ‘*the ship*’ share a common referent in our example excerpt, an anaphoric link would be annotated from ‘*the ship*’ to ‘*Cole*’. Therefore, singleton clusters (those comprising one referential mention whose referent

is not mentioned again in a document) are not annotated in MUC. That is, not all mentions which are referential are annotated; annotation depends on the context of a mention.

The guidelines note that the identity relation should be assumed to be symmetric and transitive. That is, if ‘*Cole*’ and ‘*the ship*’ are marked as coreferential, and ‘*the ship*’ and ‘*its*’ are also, it should be assumed that ‘*Cole*’ and ‘*its*’ share their referent too. However, for the purposes of evaluating system performance, the guidelines suggest that coreference be marked between a mention and its most recent antecedent mention in the document.

Noun mentions were explicitly distinguished from verbs and clauses in the guidelines. In particular, gerunds were considered markables if they were noun-like (could be modified by an adjective or take a determiner, e.g. ‘*the buying*’) but not if they were verb-like (could be modified by an adverb or take an object, e.g. ‘*buying shares*’). Expanding the range of mention forms was proposed as future work for the task.

Each mention had two spans annotated: the minimal span consisted of the head of a noun phrase (e.g. ‘*ship*’ in ‘*the ship*’) or the full name of a proper name mention (e.g. ‘*Cole*’ in ‘*the US Navy destroyer Cole*’, but the full span in the name ‘*Haden MacLellan*’), while the maximal span included the full noun phrase, including determiners and modifiers. This decision was made so as to separate the tasks of coreference resolution and syntactic parsing, and a system could get full credit for labelling coreference between minimal spans.

Noun phrases were considered analysable while entity names were considered atomic regardless of any internal structure. This means that possessive pronouns used as determiners were markables, as were nominal modifiers including tokens in noun compounds such as ‘*aluminium*’ in ‘*aluminium siding*’. However the name ‘*Iowa*’ in ‘*Equitable of Iowa Cos.*’ is not a markable due to the atomicity of the organisation name.

One implication of the decision to annotate nouns in compounds was that unquantified, bare nouns, which are typically non-referential (Non-Ref in Table 2.1), are

markables. If a bare noun referred to a type, such as the type of material ‘*aluminium*’, it was coreferential with other mentions of that type. However, if a bare noun referred to a set, such as a group of ‘*teachers*’, it was only coreferential with other references to a set containing exactly the same collection of entities. Being able to distinguish when a bare noun referred to a type or to a set was found to be difficult, which was problematic since how the decision was made could impact whether a given link was annotated.

The guidelines identify metonymy and referential ambiguity as difficult cases for annotation. Metonymy was handled by stipulating that instances be annotated according to the interpretation after the metonymy had been resolved. For instance, ‘*White House*’ should be interpreted as a reference to the presidential administration in ‘*The White House announced ...*’. Anaphoric links could additionally be labelled as optional, if a human might feasibly not be certain that identity of reference holds.

**Agreement** Inter-annotator agreement on the MUC-6 dataset was the topic of Hirschman et al. (1997). Agreement was measured (after the shared task data was annotated) by having two annotators label an initial set of five documents to identify problematic cases for discussion, before being given a further five documents to label to generate official agreement statistics.

However, agreement was similar between the two rounds of annotation, with the annotators achieving F scores of 83% on the first round and 84% on the second round. The reason for the low agreement was determined to be a problem of identifying markable spans. In particular, date spans and spans referring to less prominent entities were found to be easy to miss. After a two stage process of agreeing on markable mentions before annotating coreference, the F score of agreement improved to 91%.

Hirschman et al.’s recommendation for improving the annotation guidelines was to distinguish between extensional and intensional mentions. For instance, in the text ‘*Mr. Dooner was appointed as CEO*’, the name ‘*Mr. Dooner*’ (an extensional reference to a particular individual) should be distinguished from a role that he holds for an

undetermined period of time ‘CEO’ (an intensional predication) since failure to do so breaks the assumption of transitivity: different people may have been the CEO of Mr. Dooner’s company across time. Intensional predications include attributive mentions such as ‘*the president*’ in ‘*Barack Obama is the president*’. Here, ‘*the president*’ attributes presidency to Barack Obama, rather than referring to the man himself.

This argument was taken up further by Van Deemter and Kibble (2000), who argue that, since the motivation for defining the MUC guidelines is that two mentions are to be considered coreferential if and only if the real-world entity they refer to is the same, it is underspecified with respect to how to handle non-referential mentions. Intensional usages do not point to an entity but, rather, attribute some property to an extensional usage. In the following example, the given prices attribute numerical values to ‘*The stock price*’. Yet, annotating each as coreferential with ‘*The stock price*’, yields the contradiction that the two prices are coreferential under transitivity.

**The stock price<sub>A</sub>** fell from **\$4.02<sub>B</sub>** to **\$3.85<sub>C</sub>**.

Van Deemter and Kibble’s (2000) proposed strategies for dealing with these are:

- annotate according to the present (only annotate *C* as coreferential with *A*, if it is the current value of the stock price); or
- having attributes be a function of the seed mention and some variable e.g. time (i.e. introduce a functional, *f* which takes a time and outputs *A* or *B* according to this input); or
- exclude attributives since they are not referential (neither *B* and *C* are markables and *A* participates in no coreference relationships in this sentence).

### 2.1.2 Automatic Content Extraction

The ACE program (Doddington et al., 2004) was initiated in 2000 to extend from MUC and encourage the development of systems which could automatically extract

Dataset	Train	Test	Words	Mentions	Clusters	M/C
ACE-2	130	29	270,000	2,630	1,148	2.3
ACE03	74	31	150,000	3,106	1,340	2.3
ACE04	90	38	350,000	3,037	1,332	2.3
ACE05	57	24	350,000	1,991	775	2.6

Table 2.3: Coreference annotation statistics for the (English) ACE corpora. The number of words is as reported by Doddington et al. (2004) and NIST (2005).

knowledge about entities and events from natural language data. In particular, the organisers saw ACE systems being applied to creating a database of what is happening in the world: “who is doing what, where and when”.

In line with this motivation, coreference resolution was framed as an aspect of the target capability of Entity Detection and Tracking (EDT). In Phase 1 of EDT, a system would mark all entity references in a document and, for each mention, the entity type being mentioned. From Phase 2 in 2002, the relationships between mentions became part of EDT, with identity of reference annotated between mentions.

**Dataset** The ACE program annotated datasets<sup>4</sup> in three languages, namely English, Mandarin Chinese, and Standard Arabic; we focus on the English datasets here. In addition to newswire, documents also came from broadcast news and newspapers, for which manually and automatically transcribed versions were available. In particular, broadcast news was processed with automatic speech recognition (ASR) and newspapers with optical character recognition (OCR). In 2005, the scope increased to include weblogs and newsgroups.

Statistics for the four releases are given in Table 2.3. Note that the number of words is reported for ACE, rather than the number of tokens for MUC and OntoNotes, which means corpus size can only be compared approximately. We can see that, compared

<sup>4</sup>ACE-2: <https://catalog.ldc.upenn.edu/LDC2003T11>  
 ACE 2003: <https://catalog.ldc.upenn.edu/LDC2004T09>  
 ACE 2004: <https://catalog.ldc.upenn.edu/LDC2005T09>  
 ACE 2005: <https://catalog.ldc.upenn.edu/LDC2006T06>

to MUC, more documents have been annotated but these comprise fewer mentions and, on average, smaller entity clusters. Both are a consequence of the formulation of EDT: only mentions of entities (cf. types and concepts) are annotated, and entities can be annotated when only mentioned once in a document (i.e. a mention need not be coreferential with another to be a markable).

**Annotation** Entity Detection and Tracking annotated proper name, nominal, and pronominal references to entities from up to seven semantic classes (Linguistic Data Consortium, 2008). Initially, the classes included were Person, Organization, Facility, Geo-Political Entity, and Location; this was expanded in 2004 to include Weapon and Vehicle. Each semantic class had a fixed system of sub-classification to capture such information as an Organization being Commercial or Religious, and a Person reference being to an Individual or a Group. Unlike in MUC, coreference was encoded at the entity level, with all mentions of a given entity being labelled with the same entity identifier. That is, *'The battered US Navy destroyer Cole'*, *'its'*, *'its'*, *'the ship'*, and *'the crippled ship'* would be identified as coreferential by being assigned the same entity identifier. Coreference within a document was annotated in all datasets, and cross-document coreference in ACE 2008 (Linguistic Data Consortium, 2008).

In addition to the sub-classification of entity types, ACE introduced a classification scheme to capture the different types of mentions seen. The categories were proposed to address shortcomings noted in the MUC annotation effort and expanded by Van Deemter and Kibble (2000). In particular, a mention was labelled as one of: specific, generic, attributive, negatively quantified, or underspecified; all mentions in our Cole example would be labelled as specific mentions. In this way, intensional information (such as a person's role) could be tagged as attributive to distinguish it from specific, extensional usage. Indeed, attributive was the suggested tag for information expressed in copula constructions. Additionally, references to a particular set of entities

were labelled as specific, enabling them to be distinguished from references to types of entities, which were labelled as generic.

In a similar vein, a metonymy relation was introduced to handle this problematic case, which was noted to frequently be used when an Organization was referred to by the Facility it operated from (e.g. ‘*White House*’ for the US presidential administration).

**Agreement** Inter-annotator agreement was measured in a pilot phase of annotation, and monitored throughout the annotation process (Doddington et al., 2004). In the initial, pilot phase all documents were triple annotated, with annotators achieving F score 86%. While this is lower than the 91% reported for MUC-6, the annotation guidelines are more complex and the comparison dataset larger.

Reported cases of error include differences in ‘judgement calls’, notably on cases of referential ambiguity and where knowledge beyond what was expressed in the document was required to resolve reference. Long documents were found to be more difficult to annotate than shorter documents, with the possibility of missing coreference links between mentions appearing far apart. Additionally, there were reported errors due to the annotation tool interface and ambiguities in the annotation guidelines.

### 2.1.3 OntoNotes

OntoNotes (Hovy et al., 2006) is a large corpus with rich, cross-layer semantic annotations: each document in OntoNotes is annotated with part-of-speech tags, named entity labels, constituency parse trees, propositional structure, word sense, and coreference. While word sense and propositional labels do not have 100% coverage, the creators expect that the majority of ambiguous terms and verbal propositions are labelled.

The Conference on Natural Language Learning organised shared tasks for coreference resolution using the OntoNotes data in 2011 (Pradhan et al., 2011) and 2012 (Pradhan et al., 2012) to address difficulties in gauging the state of the art for the task. In particular, unrestricted coreference resolution had not been evaluated since the small

Genre	Train	Dev.	Test	Tokens	Mentions	Clusters	M/C
broadcast conversation	283	59	55	203,628	25,988	5,898	4.4
telephone conversation	110	16	16	103,587	15,346	2,461	6.2
bible text	319	24	26	243,040	48,636	7,695	6.3
weblogs	173	25	24	169,628	16,307	3,906	4.2
newswire	745	88	89	488,935	43,874	11,925	3.7
broadcast news	763	91	93	335,657	28,103	8,043	3.5
magazine text	409	40	45	197,520	16,226	4,293	3.8
Total	2802	343	348	1,631,995	194,480	44,221	4.4

Table 2.4: Coreference annotation statistics for (English) OntoNotes 5.

scale experiments in MUC. It was also seen that, despite being introduced to promote richer modelling of document meaning, coreference was still reliant on surface and shallow semantic features, such as gender and linguistic number.

**Dataset** As for ACE, OntoNotes data<sup>5</sup> is multi-lingual, with annotations available in English, Mandarin Chinese, and Standard Arabic, but we focus on the English release here. As well as newswire, broadcast news, and weblogs which were studied in ACE, OntoNotes includes documents from broadcast news conversation, telephone conversation, New Testament Bible text, and magazines. Long texts from the introduced genres were split into parts to facilitate annotation. All conversation text is from manual transcriptions rather than automatically processed audio files.

For the shared task, both the gold standard OntoNotes annotations, as well as the output of automatic processing for non-coreference layers, was released. Automatic annotations were generated by the BBN’s *IdentiFinder* for NER, Charniak re-ranking parser (Charniak and Johnson, 2005) for syntactic structure, *ASSERT* (Pradhan et al., 2004) for propositional structure, and an in-house tool (see Pradhan et al., 2011) for word senses. Official scores pertain to this automatic preprocessing.

<sup>5</sup><https://catalog.ldc.upenn.edu/LDC2013T19>

The corpus statistics for OntoNotes are given in Table 2.4. We can see that this corpus is at least an order of magnitude larger than previously available datasets. Indeed, the number of mentions is two orders of magnitude larger than the ACE corpora; this makes sense given that annotation is not restricted by the semantic class of the mention.

The mean number of mentions per entity cluster again sits around 4, as it did for MUC, since entity clusters comprising a singleton mention are again not annotated. Newswire and broadcast news make up just over half the dataset, but entity clusters in these genres contain fewer mentions on average. Telephone conversation and Bible text contain larger entity clusters, due to repeated chains of first and second person pronouns in the first case and of divine entities in the second (Pradhan et al., 2011).

**Annotation** Annotation decisions for OntoNotes reflect a balance between ease of annotation (to allow for the large scale of the corpus) and being consistent enough to make coreference resolution a feasible computational task (BBN Technologies, 2012).

Apart from having no restriction by semantic type, the most salient difference between OntoNotes and ACE is that OntoNotes has no sub-typing on mentions: only specific mentions are markables. Generic and underspecified mention are only markables when they are coreferential with a specific mention. However, there is no explicit rule about what constitutes a specific or a generic mention, though it is stipulated that unquantified plurals, indefinite nominals, expletive pronouns, as well as certain usages of ‘*you*’, are always generic.

ACE’s attributive category partly maintained by OntoNotes coreference relationships being sub-typed as identity of reference or apposition. However, the subject complement in copula constructions is no longer a markable, since inferring the relationship between the phrases is considered to be straightforward to derive from the syntactic structure

Annotation proceeds from the gold standard syntactic parse layer in OntoNotes. In particular, annotators are presented with noun phrases as their base markable units for

labelling. In the case of nested noun phrases with the same head word, the longest span is presented. While this removes the burden of identifying mentions from annotation, it conflates the problems of mention span detection and coreference resolution since a mention is considered correct if and only if its span matches the annotated span.

There are two cases where an annotator may add a span to those derived from the gold parse trees. Single token heads of a verb phrase may be marked as coreferential with a noun mention, as in:

Sales of passenger cars **grew** 22%. **The strong growth** followed ...

Proper name spans can be added within noun phrases markables, provided they are not adjectival. Therefore ‘*FBI*’ is a markable in ‘*the FBI spokesman*’, but ‘*US*’ is not in ‘*the US spokesman*’, since the latter is presumably equivalent to ‘*the American spokesman*’. However, proper names are still atomic and sub-spans are not markables. Therefore ‘*Massachusetts*’ is not a markable in ‘*Massachusetts Institute of Technology*’.

One case of metonymy is explicitly mentioned in the guidelines: references to a geo-political entity’s government are coreferential with the geopolitical entity itself. For instance, ‘*Lebanon*’ would be coreferential with both ‘*Beirut’s government*’ and ‘*Beirut*’ since all three refer to the geo-political entity Lebanon.

**Agreement** Annotation consistency is reasonably stable across genres, with broadcast conversation and weblogs surpassing the benchmark ‘90% solution’ (Hovy et al., 2006) at 93.7% and 91.2% (after adjudication), and broadcast news, newswire, and magazine text falling just short at 89.4%, 88.3%, and 88.8%.

To understand the problems for coreference annotation, 15000 disagreements were categorised. It was found that roughly 25% of cases represented genuine ambiguity for human readers. As well as referential ambiguity, 11% of cases involved annotators disagreeing about whether a mention was specific or generic and, so, whether it was a markable. 8% of disagreements stemmed from ambiguity in the annotation guidelines,

8% were a byproduct of the annotation tool, while the remainder were either small classes comprising less than 5% the error, or could not be categorised.

## 2.2 Evaluation

For each of the above shared tasks, a standard evaluation metric was defined and a reference implementation released to enable the quality of system outputs to be assessed and the state of the art for the task to be defined. However, due to documented biases in the proposed metrics, the problem of how to score coreference output has been addressed beyond the definition of shared tasks. There exist at least five evaluation metrics in wide use, namely the MUC score (Vilain et al., 1995),  $B^3$  (Bagga and Baldwin, 1998), CEAFM and CEAFE (Luo, 2005), and BLANC (Recasens and Hovy, 2011). These will be explained in the sections below with reference to the example output in Figure 2.1.

To address this diffusion, official CoNLL evaluation reports performance on all five metrics, though the official score is the mean of the MUC,  $B^3$ , and CEAFE scores. While this solution is a reasonable choice for evaluating a shared task, it has various shortcomings when applied to identifying promising research directions. We therefore conclude this section by introducing Kummerfeld and Klein’s (2013) solution of error-driven analysis of system output.

Gold : { ‘The battered US Navy destroyer Cole’<sub>A</sub> ← ‘its’<sub>B</sub> ← ‘the ship’<sub>C</sub> ← ‘its’<sub>D</sub> }

System : { ‘The battered US Navy destroyer Cole’<sub>A</sub> ← ‘its’<sub>B</sub> }, { ‘the ship’<sub>C</sub> ← ‘its’<sub>D</sub> }

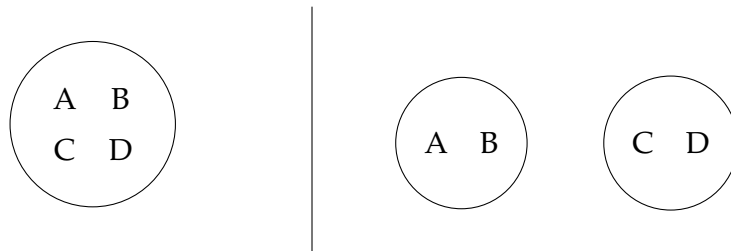


Figure 2.1: Example output for evaluation.

### 2.2.1 MUC

Vilain et al. (1995) formulate the evaluation metric used to score the MUC shared tasks. It is classified as a *link-based* metric in that it is factored over the number of links between mentions which are missing from or extra in the system output, with respect to the annotated gold standard. For instance in Figure 2.1, the system output contains two separate entity clusters which require at least one additional link between them to be equivalent to the four-mention gold cluster. Note that when coreference is annotated as a series of pairwise links between mentions, as it is in MUC corpora, entity clusters are produced by taking the transitive closure over these links.

The metric is calculated by drawing a contribution from each of the entity clusters in the gold and system output using the equations below. In particular,  $G$  refers to a gold cluster,  $S$  to a system cluster, and  $p(G)$  and  $p(S)$  to the clusters in the opposite output covering the mentions in  $G$  and  $S$ , respectively. In Figure 2.1, when  $G$  is the four-mention cluster containing mentions A, B, C, D,  $p(G)$  contains the two mention clusters since both are required to cover the four mentions in  $G$ . On the other hand, when  $S$  is either of the two system clusters,  $p(S)$  is the four-mention  $G$  cluster, even though it comprises more mentions than either of the  $S$  clusters, since it is the cluster required to cover A and B or C and D. Contributions with gold clusters as the base truth add to recall since these measure the number of gold links which are missing from system output; those with systems clusters as the base add to precision since these measure the number of spurious links in the system output.

In our example, the recall contribution reflects that a link is missing from the system output: 2 system clusters are required to cover the 4 gold mentions, hence  $R = \frac{4-2}{4-1} = \frac{2}{3} = 0.67$ . The precision contribution of each of the system clusters reflects that no links are spurious: only 1 gold cluster is required to cover the 2 system mentions, hence  $P = \frac{(2-1)+(2-1)}{(2-1)+(2-1)} = 1.00$ . The overall F score is then  $\frac{2 \times 0.67 \times 1.00}{1.00 + 0.67} = 0.80$ .

$$R = \frac{\sum(|G| - |p(G)|)}{\sum(|G| - 1)}$$

$$P = \frac{\sum(|S| - |p(S)|)}{\sum(|S| - 1)}$$

$$F = \frac{2PR}{P + R}$$

**Biases** Bagga and Baldwin (1998) identify two shortcomings with the MUC metric. Firstly, since the metric is link-based, it does not include clusters of singleton mentions explicitly in its calculations. This means that systems do not get any credit for correctly identifying which mentions in a document are not coreferential with any other. There are two considerations when thinking about this shortcoming. On the one hand, discourse singletons are the majority class when annotating coreference, and we still want our metric to be discriminative with respect to how well systems annotate coreference relationships. However, the task of classifying whether a mention is a discourse singleton or not is very difficult (e.g. Ng and Cardie, 2002a; Uryupina, 2003; Ng, 2004; Recasens et al., 2013), and scoring should reflect how well a system can perform this classification.

Bagga and Baldwin’s second criticism of the MUC metric is that the metric is blind to how damaging a coreference error is to output. In particular, they argue that errors concerning large entity clusters (which presumably relate to topical discourse entities) are more damaging than errors concerning smaller ones. Recasens and Hovy (2011) further this criticism by considering it from the point of view of the number of links involved in an incorrect decision. The reason they say that errors involving larger clusters are worse is because the *total* number of involved links is larger and, by being formulated around the *minimum* number of links required to repair output, MUC does not capture this insight. Predominately for this second reason, Bagga and Baldwin define a new metric to evaluate coreference output, B<sup>3</sup>.

### 2.2.2 $B^3$

In contrast to the MUC metric which is defined over coreference links,  $B^3$  is calculated by iterating over the mentions in a dataset. In this way, it is possible for entity clusters which contain more mentions to contribute more to the final score.

The contribution from each mention is given in the following equations, which are designed to capture the purity of the entity cluster the mention has been assigned to in the system output. In particular, the numerator of the mention's contribution to both precision and recall comes from the number of mentions in the system entity cluster which are coreferential in the gold standard (including self-links), while the denominator is the size of the system entity cluster for precision and the size of the gold entity cluster for recall. In this way, recall reflects what proportion of the gold cluster is captured by the system cluster and precision reflects what proportion of the system cluster is correct with respect to gold.

$$R = average(\frac{|G| \cap |S|}{|G|})$$

$$P = average(\frac{|G| \cap |S|}{|S|})$$

Since each of the four mentions in our example are clustered correctly with a partner in the system output and the gold cluster has size 4, the  $B^3$  recall would be  $R = average(\frac{2}{4}, \frac{2}{4}, \frac{2}{4}, \frac{2}{4}) = 0.50$ . Likewise,  $B^3$  precision would be  $P = average(\frac{2}{2}, \frac{2}{2}, \frac{2}{2}, \frac{2}{2}) = 1.00$ . Therefore, the  $F = \frac{2 \times 0.50 \times 1.00}{1.00 + 0.67} = 0.67$ :  $B^3$  scores the clustering lower by assigning a larger penalty on recall to reflect that the missing link impacts the interpretations of all four mentions in the gold cluster.

**Variations** (Pradhan et al., 2014) highlights that the  $B^3$  metric iterates over a fixed set of mentions, which makes it undefined on mentions automatically extracted by systems. Necessarily, such mention sets will not correspond exactly to those in the

gold standard, which  $B^3$  targeted. For this reason, the literature reports a number of variations on  $B^3$  which vary in how they handle cases where either an annotated or an extracted mention cannot be aligned to one in the converse set, commonly referred to as *twinless* mentions.

Bengtson and Roth (2008) simply ignore mentions which cannot be aligned, arguing that this is justified given that mention detection has over 90% coverage. However, the following literature finds this solution overly lenient. Stoyanov et al. (2009) propose two variations of the  $B^3$  metric, namely  $B^3$ -all and  $B^3$ -0.  $B^3$ -all retains all non-aligned mentions and punishes their presence by having them contribute  $\frac{1}{|G|}$  in the case of a gold mention missing from the system output and  $\frac{1}{|S|}$  in the case of a spurious mention.  $B^3$ -0 discards all spurious extractions, but penalises recall by having all missed mentions contribute zero to recall.

Cai and Strube (2010) find these variants flawed in that  $B^3$ -all assigns credit for spurious mentions in the system output in singleton clusters, while  $B^3$ -0 does not penalise erroneous coreference relations if their mentions do not appear in the gold standard. Yet another variation was proposed in Rahman and Ng (2009): discard all unaligned spurious mentions which are singletons since the system has correctly predicted that these are not coreferential with any other mention in the document. Cai and Strube (2010) deem this valid and note that it gets around the shortcoming of  $B^3$ -all, but does not address the shortcoming of  $B^3$ -0.

To rectify this divergence of how  $B^3$  is being evaluated and, more importantly, reported for comparison, Cai and Strube (2010) proposes yet a further variant, and this was used in the evaluation of the CoNLL shared tasks. Cai and Strube’s variant draws on the ideas from the previous  $B^3$  variants, but adjusts them to produce more intuitive results. Concretely, all mentions which are missing from system output are added as singleton clusters since the system did not find them to be coreferential with any other mention in the document; all spurious mentions in the system output are

either discarded if singletons, or added to the gold standard as singletons if they were erroneously included in coreference relationships.

Pradhan et al. (2014) re-interpret how the  $B^3$  should be implemented based on communication with the original authors. That is, they resolve the underspecification of the metric on system mentions, rather than proposing a new variation. Their most recent release of the official scorer implements  $B^3$  such that recall is calculated by iterating over gold mentions and precision is calculated by iterating over system mentions, thereby not requiring any explicit mapping between the two sets.

**Biases** Luo (2005) argues that  $B^3$  is flawed by allowing the same gold cluster to be aligned to multiple different system clusters and vice versa, as we saw in our alignment in Figure 2.1. He argues that this does not allow the metric to correctly penalise systems for producing an incorrect number of clusters.

Recasens and Hovy (2011) find that  $B^3$  is highly sensitive to the number of singleton clusters there are in the mention set. In particular, as the number of singletons grows, the  $B^3$  score tends to inflate such that differences in how well the system classified coreference relationships is obscured. However, it is a difference which is most relevant for corpora which annotate singleton discourse entities, such as ACE (61% of entities are singletons), rather than those which do not, such as MUC and OntoNotes.

### 2.2.3 CEAF

Constrained Entity Alignment F scores (CEAF, Luo, 2005) seek to improve coreference resolution evaluation by ensuring that errors in the number of entity clusters produced by a system translate to penalties in score. It does this by finding an optimal alignment between gold and system clusters with the constraint that each system cluster is aligned to at most one gold cluster, and vice versa. After finding this alignment, the scores are calculated by iterating over the pair and using one of two similarity metrics to determine the correspondence between the two.

There are two variants of the metric, the mention-based CEAFM and the entity-based CEAFE, based on the similarity metric used. CEAFM iterates over mentions calculating a contribution for the purity of the cluster it is in while CEAFE iterates over entities calculating the overlap between gold and predicted entities. Specifically, to calculate CEAFM, the following equations are used.

$$R = \frac{\sum(|G \cap S|)}{\sum(|G|)}$$

$$P = \frac{\sum(|G \cap S|)}{\sum(|S|)}$$

These equations look very similar to those used to calculate  $B^3$ : the numerator and denominator are the same, but scores are combined by summing rather than averaging. The scores may also be different due to the constraint on 1-1 mapping between the gold and system output. However, although only one of our system fragments can be aligned to the four mention gold cluster, the precision and recall turn out to be the same as they were for  $B^3$ . That is  $R = \frac{2}{4} = 0.50$  and  $P = \frac{2}{2} = 1.00$ , regardless of which cluster is selected in the alignment. This is consistent with Luo's interpretation that CEAFM reflects the proportion of mentions in the correct entity clusters, since the gold cluster is split in half, without any spurious relationships introduced.

On the other hand, to calculate CEAFE, the following equations are used.

$$R = \frac{\sum(\frac{2|G \cap S|}{|G|+|S|})}{\sum(G)}$$

$$P = \frac{\sum(\frac{2|G \cap S|}{|G|+|S|})}{\sum(S)}$$

The numerators in these equations reflect an alternative measure of cluster purity to  $B^3$ , while denominators calculate the number of entity clusters in the gold standard and system output, respectively. That is,  $R = \frac{4}{6} = 0.67$  since the new purity estimate is  $\frac{2 \times 2}{4+2}$

Gold : { 'The battered US Navy destroyer Cole'  $_A \leftarrow$  'its'  $_B \leftarrow$  'the ship'  $_C \leftarrow$  'its'  $_D$  } ,  
 { 'a huge Norwegian transport vessel'  $_E$  }

System : { 'The battered US Navy destroyer Cole'  $_A \leftarrow$  'its'  $_B$  } , { 'the ship'  $_C \leftarrow$  'its'  $_D$  } ,  
 { 'a huge Norwegian transport vessel'  $_E$  }

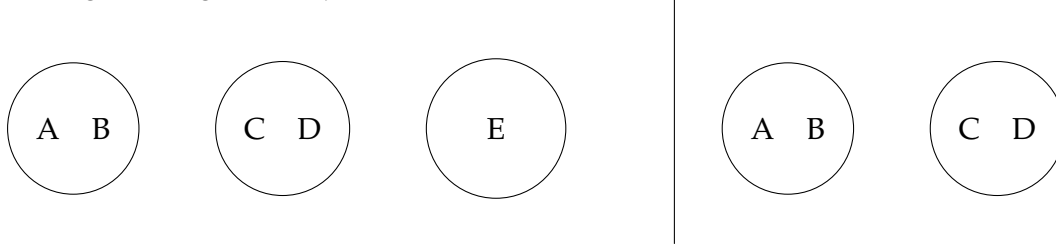


Figure 2.2: Example output for BLANC evaluation.

and there is one gold cluster, while  $P = \frac{4}{6} = 0.67$  since there are two system clusters output. The F score is therefore  $F = \frac{2 \times 0.67 \times 0.33}{0.67 + 0.33} = 0.44$ . This is the lowest score we have derived for our example and this makes sense: CEAFE is designed to measure the proportion of entity clusters which are found in both the gold and system output and, without aligning the system fragments to the same gold cluster, the alignment misses half the mentions.

**Biases** Recasens and Hovy (2011) find that the CEAF scores are just as sensitive to the number of singleton clusters as  $B^3$  scores are, which limits its ability to discriminate between the quality of various system outputs, particularly on corpora which annotate singleton entity clusters.

## 2.2.4 BLANC

Recasens and Hovy (2011) propose BLANC to provide a better spread of scores than  $B^3$  and CEAF in the case that singleton clusters are annotated. It also addresses the bias of MUC in which it underestimates the impact of errors in large clusters by only considering the minimum (vs. total) number of links involved in the error. It does this

$$\begin{aligned}
R_{coref} &= \frac{\text{correct\_coref\_links}}{\text{gold\_coref\_links}} & R_{non\_coref} &= \frac{\text{correct\_non\_coref\_links}}{\text{gold\_non\_coref\_links}} \\
P_{coref} &= \frac{\text{correct\_coref\_links}}{\text{predicted\_coref\_links}} & P_{non\_coref} &= \frac{\text{correct\_non\_coref\_links}}{\text{predicted\_non\_coref\_links}}
\end{aligned}$$

by adapting the Rand index (Rand, 1971), a metric devised to evaluate clustering, by treating clusters of mentions in coreference as clusters of nodes in the general setting.

The calculation of BLANC involves the calculation of two F scores, one to capture how accurately a system labels coreference relationships,  $F_{coref}$ , and another to capture how well the system classifies mentions as singletons,  $F_{non\_coref}$ . All mention pairs in a corpus contribute to one of the F scores. If there is no coreference relationship between the two, the link contributes to  $F_{non\_coref}$  and if there is, it contributes to  $F_{coref}$ . These scores are simply defined as the proportion of correct links, normalised by the number of gold links to give recall (the proportion of gold links retrieved) and by the number of system links to give precision (the proportion of system links that are correct). F scores are the standard harmonic mean of the corresponding precision and recall.

We use the modified example in Figure 2.2, which includes an additional mention which is not coreferential with any in the gold cluster of Figure 2.1 to better illustrate how BLANC is calculated. We note that in OntoNotes, this mention will only be included in evaluation if it is part of a larger, non-singleton cluster, though the steps involved in calculating BLANC proceed in the same way as described here.

There are now 10 mention pairs in the example, one from each mention to each of the mentions preceding it; in the gold standard, 6 of these mention pairs are coreference relations and 4 are non-coreference relations. In the system output, 2 coreference relationships are identified and both are correct ( $A \leftarrow B$  and  $C \leftarrow D$ ), giving  $R_{coref} = \frac{2}{6} = 0.33$ ,  $P_{coref} = \frac{2}{2} = 1.00$ , and  $F_{coref} = 0.50$ . However, the system output labels 8 mention pairs as non-coreferential, where only 4 ( $A \leftarrow E$ ,  $B \leftarrow E$ ,  $C \leftarrow E$ , and

$D \leftarrow E$ ) are in the gold standard. Therefore our example scores  $R_{non\_coref} = \frac{4}{4} = 1.00$ ,  $P_{non\_coref} = \frac{4}{8} = 0.50$ , and  $F_{non\_coref} = 0.67$ .

In the standard setting, the arithmetic mean of these two scores produces the final assessment of system quality,  $BLANC = avg(0.50, 0.67) = 0.57$ . In giving each equal weight, BLANC is designed not to be inflated, thereby less discriminative to the coreference classification problem, in corpora which annotate singleton clusters.

### 2.2.5 CoNLL

Despite the availability of multiple, motivated evaluation metrics, the problem of how to best evaluate coreference resolution output in a way that is both intuitive and unbiased remains an open question. One undesirable outcome of this proliferation of metrics is that, due to biases in each, they make different assessments about whether one system's output is better than another's. This shortcoming is exemplified in Table 2.5, which shows the scores assigned to all systems submitted to the CoNLL-2012 shared task by Pradhan et al.'s (2014) reference implementation from the official scorer.

These scores are sorted by their CoNLL score, the official metric for the task, which averages a system's MUC, B<sup>3</sup>, and CEAFE scores. We can see that this score provides a fair indication of the relative quality of each system, making it suitable for the purposes of scoring the shared task. That is, while the official ranking obtained from the CoNLL score is not reflected uniformly across all metrics, there are only a few and small exceptions which break the trend, and these are indicated in boldface and underline.

### 2.2.6 Error-Driven Evaluation

It is problematic that the different metrics make different assessments about output quality when translating work to the research space, given that systems typically are designed to optimise for the evaluation metric and it is this optimisation process which drives future research directions. Kummerfeld and Klein (2013) also question the use of 'monolithic' scores being used to assess system quality for research since such metrics,

Entrant	MUC	B <sup>3</sup>	CEAFM	CEAFE	BLANC	CoNLL
fernandes	70.51	57.58	61.42	53.86	58.75	60.65
martschat	66.97	54.62	58.77	51.46	55.04	57.68
bjorkelund	<b><u>67.58</u></b>	54.47	58.19	50.21	<b><u>55.42</u></b>	57.42
chang	66.38	52.99	57.10	48.94	53.86	56.10
chen	63.71	51.76	55.77	48.10	52.87	54.52
chunyang	63.82	51.21	55.10	47.58	52.65	54.20
stamborg	<b><u>64.26</u></b>	51.66	55.10	46.60	<b><u>54.42</u></b>	54.17
shou	62.92	49.44	53.16	<b><u>46.66</u></b>	50.44	53.00
yuan	62.55	<b><u>50.11</u></b>	<b><u>54.53</u></b>	45.99	52.11	52.88
xu	<b><u>66.18</u></b>	<b><u>50.30</u></b>	51.31	41.25	46.47	52.58
uryupina	60.89	46.24	49.31	42.93	46.04	50.02
songyang	59.84	45.90	<b><u>49.58</u></b>	42.36	45.10	49.36
zhekova	53.52	35.66	39.66	32.16	34.80	40.45
xinxin	48.27	<b><u>35.73</u></b>	37.99	31.90	<b><u>36.54</u></b>	38.63
li	<b><u>50.84</u></b>	32.29	36.28	25.21	31.85	36.11

Table 2.5: Official scores of the competing systems at CoNLL-2012.

by design, give an overview of system quality, abstracting over the particular errors seen in system output. Specifically, single scores give little insight into promising avenues of future work or the cascading impact of a single error.

To provide a finer-grained alternative, Kummerfeld and Klein proposes a procedure which reports on the repairs which need to be applied to the system output to transform it to the gold standard. The seven error categories reported are span error, missing entity cluster, spurious entity cluster, missing mention, spurious mention, divided entity cluster, and conflated entity cluster. Our example in Figure 2.1 shows an instance of divided cluster.

These error categories are tallied by a process of aligning gold and system mentions and comparing the pairings using a pipelined five-stage classification process.

- 1) correct system mention spans to match those annotated in the gold standard

- 2) split system clusters which are not homogeneous with respect to the gold to be such
- 3) remove spurious mentions from system output
- 4) insert missing mentions into the system output
- 5) merge system fragments to achieve the gold clustering

The raw numbers of these repairs are converted heuristically to the seven reported error classes.

## 2.3 Summary

We have seen that, while coreference is an intuitively simple concept, there have been numerous challenges in defining it as a computational task and evaluating the quality of a system. While mainstream annotation efforts, mostly recently OntoNotes, have produced large datasets on which systems can be developed and evaluated in a standard setting, referential ambiguity has been a trouble case in the annotation of each. This manifests itself in reduced inter-annotator agreement statistics on mentions of complex entities, particularly nominal mentions. In this thesis, all benchmarking will conform to CoNLL standards on the English portion of the English OntoNotes corpus. To better understand areas for improvement, we additionally use the error analysis toolkit of Kummerfeld and Klein (2013).

In the next chapter, we will review approaches to the task set out by annotation guidelines, as well as the relative performance of systems competing at the shared tasks. This review is then used to motivate the design of the coreference resolution system whose development is a key contribution of this thesis.

## 3 Background

Coreference resolution, and reference resolution more generally, is an important component of natural language processing pipelines. Resolving that linguistic expressions refer to mutually understood entities and concepts, and that different linguistic expressions may refer to the same entity (and, conversely, that the same expression can refer to different entities) is necessary for humans and automatic systems to understand the meaning being expressed in a discourse. Modelling how humans resolve reference has been explored in the linguistic literature and this frames our review of computational approaches to coreference resolution in this chapter.

In Section 3.1, we survey the models which have been proposed for coreference resolution. This section starts with an overview of cognitive models of linguistic reference, which guides the following discussion of computational models. In Section 3.2, we then enumerate the list of coreference indicators described in the literature, giving first their description in linguistic and cognitive theories on anaphora resolution, which then motivates their implementation as features in coreference engines.

Our survey shows that linguistic insights have been effective in pushing forward the state of the art, but there remain insights which have yet to be explored for improving computational models for coreference resolution.

### 3.1 Models

Coreference is typically defined in shared task guidelines as identity of reference between noun mentions. Because of this, our first goal is to understand how reference

is created by language. The review of linguistic models below demonstrates that reference is a relationship between linguistic expressions and abstract discourse entities which take shape as a discourse proceeds. While this is inconsistent with shared task guidelines which are based on real world referents, the accompanying theories offer us insight into how humans create and track discourse entities and their relationships, which is useful for understanding what properties our computational models should have. We use this description as the basis for the design of our system in Chapter 4.

We then survey the computational approaches which have been applied to the task. We find that entity-level models are consistent with linguistic models of reference but are generally not competitive with the more widely used mention-pair model. However, the recent success of structured prediction approaches to coreference has argued for the benefit of enriching mention-pair models with entity-level features.

### 3.1.1 Linguistic Reference

The problem of understanding linguistic reference can be thought of as modelling the objects which stand as the referents of linguistic expressions and how humans resolve linguistic expressions to these referent objects. For instance, reading the linguistic expression ‘*The US Navy destroyer Cole*’ will cause a human to draw to mind some representation of the warship and we would like to understand what is the nature of the object drawn to the reader’s mind and what drives this process. The first question is addressed in this section, while the second is the topic of Section 3.2.1.

While the early literature on the representation of discourse entities used logical objects (e.g. Russell, 1905), current theories hold referents to be cognitively based objects with properties that are informed by the linguistic context of their mentions. For instance, Heim (1982) likens referents to *file cards* which are created when an entity is introduced into a discourse and retrieved upon subsequent mention of their referent. A series of file change semantics can apply to the file card as more information about the referent is revealed in the discourse.

Mental space theory (Fauconnier, 1994) is likewise incremental in that the properties of cognitive objects are populated as they are revealed in their discourse. However, the nature of referent objects is recast with each referential expression creating a new mental space object, which may be related to other mental space objects in any number of ways. The identity relation is the target for coreference resolution, and is indeed highlighted as the vital relation for facilitating communication in that it enables language users to refer to an entity multiple times and have its narrative be construed as continuous. The important aspect of this theory for our work is that there is no one single object representing a discourse entity, but as many objects as there are referential expressions, each related via identity.

Fauconnier and Turner (2008) expand mental space theory by describing the processes in which spaces are understood to be identical, namely *blending* and *compressing*. Blending is the process by which human imagination likens dissimilar mental space objects to one another via analogy. Compression is the process of omitting certain properties so that incongruous mental spaces can be seen as similar. For instance, in the following sentence from part 1 of the OntoNotes document ‘*cnn\_0007*’, the mental space objects associated with ‘*White House*’ and ‘*the administration*’ can be identically related by compressing the fact that ‘*White House*’ refers to a physical location.

... But her husband being a prominent **White House** critic who clearly **the administration** was angry at and wanted to.

Mental space theory relaxes the requirement for mentions to cluster only according to discourse entities, instead mentions are able to relate to one another flexibly. While this fuzziness gives the model rich descriptive power, it is more powerful than the picture of coreference resolution given by shared task guidelines.

Versley (2008) argues that the fuzziness of cognitive models is inherent and this is what causes referential ambiguity — uncertainty about whether two related linguistic units are coreferential. Entities are multi-faceted and may be referred to in many different frames of reference, with a frame of reference selecting among these different

facets. Near identity of reference is introduced by Recasens et al. (2011) to explain referential ambiguity. In the proposed model, linguistic expressions are encoded as mental space objects, whose properties reflect the information expressed in the linguistic context of the expression. Referents of related linguistic expressions fall along a continuum with identity of reference at one extreme and non-identity at the other. Expressions need not refer exactly to the same discourse entity to be considered by a human as coreferential; reference merely needs to be near enough. Specifically, resolving coreference, and the lack thereof, comes from the processes of refocusing and neutralisation, which are akin to Fauconnier and Turner’s blending and compression.

We take the above cognitive models as our underlying theory of reference in this thesis. However, while we will assume that discourse entities correspond to fuzzy psychological objects, we also believe that shared task guidelines capture human intuition to a reasonable degree. In particular, we see them as an approximation of reality. Therefore, we explore how the state of the art for coreference resolution can be extended, but also how referential ambiguity impacts the performance of our system.

With such a model in mind, we now review a representative selection of the computational approaches which have been applied to coreference resolution, starting with those used at the 7th Message Understanding Conference (MUC-7); for a review of work pre-dating shared tasks, see Mitkov (1999). These early rule-based systems were heavily informed by cognitive theories of reference, though were limited by having small feature sets due to the small scale of the MUC corpora. With larger datasets, machine learning approaches with rich feature sets could be applied to the task. The standard way to cast coreference resolution as a machine learning problem is the mention-pair model. While such systems produce promising results, the current trend is to reintegrate the notions of incremental processing and emergent discourse entities into machine learning approaches. The work in this thesis represents a contribution to this effort, which we argue more faithfully represents cognitive theories.

### 3.1.2 Rule-Based Discourse Models

Five teams competed in coreference resolution at MUC-7, namely OKI (Fukumoto et al., 1997), and the Universities of Durham (Garigliano et al., 1998), Manitoba (Lin, 1998c), Sheffield (Humphreys et al., 1998), and Pennsylvania (Baldwin et al., 1998), with all but the University of Pennsylvania using systems designed to compete in multiple task tracks. There is no detail about the model used by the OKI system and very little about the University of Pennsylvania system in the shared task reports. The following discussion therefore omits the OKI system and bases the discussion of University of Pennsylvania on Baldwin (1997).

Overall, the design of the resolution module of the four university systems is relatively simple and there are many similarities between the submissions. In particular, all maintain a store of discourse entities, the *discourse model* of a document, and this is incrementally populated and updated as a document is processed. A document is processed by iterating over the extracted mentions in left-to-right, top-to-bottom reading order, with each triggering a search for candidate antecedents among the entities in the discourse model. If a compatible antecedent is found, the new mention is added to the entity and its attributes updated according to the linguistic form and context of the mention; otherwise, a new entity object comprising just the current mention is added to the discourse model. For Pennsylvania and Durham, all compatible candidates indicated by these features are returned and then sent to a second stage which selects the most salient candidate based on grammatical and semantic features, position in sentence, mention recency, and relatedness to the topic of the text. On the other hand, Sheffield's and Manitoba's indicators are each assigned a relative confidence and this is used to rank candidate antecedents, with the best selected.

This approach is linguistically licensed in that discourse entities take shape as a document is processed, in a similar way to how a human reader might process the

System	MUC-7		
	R	P	F
OKI	28.4	60.6	38.6
Durham	46.9	57.0	51.5
Pennsylvania	46.8	78.0	58.5
Manitoba	58.2	64.2	61.1
Sheffield	56.1	68.8	61.8

Table 3.1: Performance of rule-based entity level models on MUC-7.

document. Indeed, Baldwin (1997) explicitly models its approach on file card semantics of Heim (1982).

The MUC score for the best performing system from each team is given in Table 3.2. Despite the similarity of their systems, there is a noticeable spread in their performance. The University of Pennsylvania substantially outperforms its competitors in precision, at the expense of relatively poor recall. The Manitoba and Sheffield Universities strike more of a balance between precision and recall, and achieve the best results overall. Given the similarity of their system descriptions, we can infer that implementation decisions are an important factor for model performance.

We note that Pennsylvania report that their system often finds an antecedent for a common noun mention which seems to be acceptable to its developers, but which has not been labelled in the dataset. This could be a limitation of the annotations in MUC, or a consequence of the referential ambiguity problem described above.

### 3.1.3 Mention-Pair Models

In order to cast coreference resolution as a learning problem, researchers typically formulated it as a binary classification task between mentions. That is, given a pair of mentions extracted from a document, a classifier is trained to assign the positive class to pairs which belong to the same coreference cluster, and the negative class to pairs

System	MUC-6	MUC-7	CoNLL-2011	CoNLL-2012
Soon et al. (2001)	62.6	60.4		
Soon et al. reimplementations	66.3	61.2		
Ng and Cardie (2002b)	67.5	63.0		
Reconcile	71.2	62.9	51.92	
UIUC			55.96	
Yang et al. (2003)	71.3	60.2		
Finkel and Manning (2008)	68.3			
Durrett and Klein (2013)			60.13	61.79
Wiseman et al. (2015)				63.39

Table 3.2: Performance of mention-pair models on standard evaluation corpora.

which do not. Such models are commonly referred to as mention-pair models, and remain competitive in the current research space.

However, the model has been criticised for its independence assumptions. In classification, each mention-pair is processed independently of other pairings for a given mention, not allowing candidate resolutions to compete with one another. Clustering is defined as a static post-process decoding step, not allowing previous decisions to influence later ones. For these reasons, which we note contradict cognitive models of human discourse processing, mention-pair models are prone to global consistency errors. These have been addressed by using increasingly sophisticated reasoning for clustering.

### Closest First Clustering

Soon et al.’s (2001) system was designed for the MUC task definition and was the first learning based system to be competitive with the above rule-based systems. The top section of Table 3.2 shows the performance of the Soon et al.’s approach on the MUC-7 dataset; the first value is the reported performance of the original system and the second reimplementations value was produced by Ng and Cardie (2002b) using improved mention extraction and feature value calculation. The performance of

the systems is indeed comparable with MUC results, outperforming all systems but Sheffield, who won the shared task.

Soon et al.'s system uses a pipeline architecture of pre-processing before coreference resolution, and post-processing. Mentions are extracted in preprocessing, using trained HMM models which identify noun phrase chunks given POS and NER tags over tokens. Soon et al. finds that this simple method had roughly 85% coverage on a subset of the MUC-6 training documents, with many errors due to incorrect span determination rather than spurious noise.

Resolution iterates over extracted mentions by generating a series of mention pairs for the current mention with the mentions preceding it in its document and passing these instances to a C5<sup>1</sup> decision tree classifier. In training, a mention with its closest antecedent in the gold answer key constitute a positive training instance, while the mentions between the current mention and its closest antecedent each trigger negative training instances. At test time, each mention preceding the current mention is considered in turn, with the first pairing yielding a score over a predefined threshold of 0.5 being resolved to be the mention's antecedent, terminating the search process. Because of this design, decoding in Soon et al. is called *closest first* clustering. If no preceding mention is positively classified, the mention is treated as non-anaphoric, potentially starting a coreference cluster or becoming a discourse singleton. Postprocessing produces clusters by greedily chaining together compatible pairs of mentions in left-to-right reading order.

Analysis of system performance reveals that the small dataset of MUC is not problematic: the system achieves peak performance after about 25 training documents and begins to overfit the data after this point. While overfitting is a particular danger for decision tree classifiers, the small amount of training data required is surprising. It is probably explained by the system's small feature set: only 12 features are defined, and the best model uses only 8 of these.

---

<sup>1</sup><http://www.rulequest.com/see5-info.html>

### Best First Clustering

Ng and Cardie (2002b) argue that the closest first clustering of Soon et al. is insufficient to model coreference due to the presence of distractor mentions: selecting a locally acceptable antecedent without considering a range of potential candidates could mean that a better one is missed. To address this shortcoming, they introduce *best first* clustering in which decoding does not terminate upon finding a compatible antecedent for a given mention, but rather exhaustively searched among candidate antecedents for the one which is *most* compatible with the mention.

At the same time, they also use training instance selection to adjust what instances are used to train their system. Instance generation still proceeds using Soon et al.'s method of reverse iteration from the current mention but they extend the endpoint for non-pronoun mentions to the closest non-pronoun antecedent. In this way, the positive training instance generated is based on either a proper name or common noun comparison, which Ng and Cardie suggest should correspond to an informative mention pairing. They also improve Soon et al.'s string match feature by sub-classing it according to the mention types involved; different features are generated for the cases when the paired mentions are (1) both proper names, (2) both pronouns, or (3) both non-pronouns.

The combination of these three changes significantly improves performance, as seen in the second section of Table 3.2. On both datasets best first clustering substantially improves system recall while sub-classing string match improves precision, though this improvement is only large enough to impact MUC-7 F score. Training instance selection does not improve performance on either dataset when introduced without the simultaneous introduction of the other two changes.

Reconcile<sup>2</sup> (Stoyanov et al., 2010a,b, 2011) is the publicly available system which extends Ng and Cardie's work. In particular, it achieves competitive performance on modern coreference corpora with best first clustering learned using either a perceptron

---

<sup>2</sup><https://www.cs.utah.edu/nlp/reconcile/>

or decision tree classifier, presumably due to its extensive feature set (see Section 3.2). Likewise the publicly available University of Illinois - Urbana Champagne (UIUC) system<sup>3</sup> (Bengtson and Roth, 2008; Chang et al., 2011) implements a best-first mention-pair clusterer that achieves highly competitive performance, attributed to its rich feature set by its developers. Training instances are generated using the procedure described in Soon et al. and learning uses an averaged perceptron classifier.

### Competition Learning

While best-first decoding implicitly captures competition between candidate antecedents for a given mention via their relative classifier scores, the approaches we have seen score mention pairs independently of one another; Yang et al. (2003) argue that this is problematic because it does not allow candidates to compete directly with one another.

To model competition, Yang et al. develop the *twin candidate model* in which training instances are formed by the current mention, one preceding mention annotated in the same entity cluster, and one preceding mention not in the same gold cluster. The classifier is given the task of determining which of the two candidates is the correct antecedent of the given mention. This is modelled using features defined between the mention and each candidate, as well as between the candidates themselves. In particular, features for the twin comprise the distance between the competing antecedents in the document, which of the two have a more similar surface form to the current mention, and which of the two are more semantically related to the current mention. The correct antecedent for a given mention is determined in a series of twin comparisons: it is the mention which wins against the most competing antecedents.

The training instances generated are a subset of the quadratic number of possible twins. For pronoun mentions, instances are created using all mentions which agree with the pronoun in number, gender, and person from the current and the preceding two sentence context. For non-pronouns, instances are created using all non-pronoun

---

<sup>3</sup>[http://cogcomp.cs.illinois.edu/page/software\\_view/Coref](http://cogcomp.cs.illinois.edu/page/software_view/Coref)

mentions in the current, previous, and next sentences. No training instance is generated for mentions which start an entity cluster, or are discourse singletons.

Table 3.2 shows that results for Yang et al. approach are fair compared to standard, best-first models. The reported performance of Yang et al. exceeds that of Ng and Cardie on MUC-6 but trails it on MUC-7. Yang et al., however, reports adjusted performance figures for Ng and Cardie based on the two systems using the same baseline feature set, and in this evaluation, Yang et al. outperforms Ng and Cardie on both datasets (MUC-6: 69.4 vs. 71.3 and MUC-7: 58.7 vs. 60.2). While vagueness of detail limits direct comparison, there appears to be promise in directly modelling competition.

Similar to Yang et al., Denis and Baldridge (2008) criticise the independence assumptions of previous mention-pair implementations, but instead address the shortcoming by moving to a model in which *all* candidate antecedents compete synchronously. This is achieved by using a learned ranker in place of the classifier in their system. While the goal is not to learn a complete ordering over candidates, Denis and Baldridge (2007) show how a ranking architecture allows feature value determinations to be visible between different mention pairings.

The move to a ranking framework is problematic since a ranker will always output the best choice among the candidates, even when the current mention is non-anaphoric (should start an entity cluster or be a discourse singleton). The authors therefore implement a binary classifier to label mentions as anaphoric or not, and pipeline this to precede ranking. In this way, only anaphoric mentions which are considered to corefer with an earlier mention are sent to the ranker and non-anaphoric mentions are left unresolved.

Denis and Baldridge's results are not comparable with the systems reported in Table 3.2 since they evaluate on an ACE dataset but do not report ACE score. However, when a ranker is substituted in place of a classifier in their strongest system, performance increases by 1.5 CoNLL score. This is a strong result, particularly given

the reported accuracy of their anaphoricity classifier is only 80.8% (which means that nearly 20% anaphoric mentions are erroneously filtered, guaranteeing a recall penalty).

Finkel and Manning (2008) and Durrett and Klein (2013) both develop from Denis and Baldridge's proposal that clustering should be mention synchronous.

Finkel and Manning proposes an Integer Linear Programming (ILP) model for coreference resolution in which the sum of classifier scores over the graph of possible mention pairings is maximised, subject to the constraint that clusters be transitive: whenever the pairs  $(A, B)$  and  $(B, C)$  are classified to be coreferential,  $(A, C)$  should be as well. Overall, results were mixed. In particular, the ILP formulation performed worse than a standard mention-pair baseline when performance was measured with the link-based MUC score (shown in Table 3.2) but up to 2.7% better when measured with the entity-aware B<sup>3</sup> score. This suggests that while the raw number of pairwise decisions made by their ILP system was not as good as in their mention-pair baseline, the quality of the system output was better, particularly on large entity clusters.

Durrett and Klein, on the other hand, describes a publicly available state-of-the-art system<sup>4</sup> which models coreference resolution using a log-linear model. Analogous to Finkel and Manning, the likelihood of a particular resolution of the mentions in a document is taken from the sum of the ranker's scores of all possible mention pairings, together with terms to represent mentions being discourse new. This likelihood term is augmented with a loss function parameterised over three error classes, namely a mention being falsely labelled anaphoric, falsely being labelled as discourse new, and being assigned an incorrect antecedent. By defining the task in this way, mention pairs still compete via their relative scores, but the framework itself gives candidate resolutions a direct means of competing with one another. This framework is powerful and, together with novel feature design, places Durrett and Klein among the best performing approaches to coreference resolution so far reported.

---

<sup>4</sup><http://nlp.cs.berkeley.edu/projects/coref.shtml>

Post-dating the work in this thesis is Wiseman et al. (2015), who currently are credited as the best reported performance on CoNLL-2012 task. Their state-of-the-art approach again uses simple mention-pair feature, though formulate coreference resolution as the joint task of mention anaphoricity classification and reference resolution and learn these with a neural network framework. We include it here as an example of competition learning since it jointly weighs evidence for the specified sub-tasks.

### 3.1.4 Entity-Level Models

An alternative way to address the shortcomings of traditional mention-pair implementations has been to return to the early discourse model style approaches to coreference used at MUC, in which clusters are incrementally built and updated on processing each mention. In this way, comparisons are between mentions and entity clusters of resolved mentions, allowing previous decisions to influence later ones.

The problem with this model is that it is unclear how to best cast it as a machine learning problem. Early attempts were straightforward extensions of the mention-pair model in that comparisons were still between two mentions. That is, a cluster-mention comparison was modelled as a series of comparisons between the current mention and each of those in the cluster, with these pairwise scores being combined after the fact to represent compatibility. It is difficult to assess the effectiveness of this approach since, as a whole, these systems were not evaluated under standard shared task conditions. However, most fail to substantially improve when compared to a standard mention-pair baseline, suggesting that more sophisticated modelling is required.

The more recent way to leverage incremental clustering is to make comparisons explicitly between a mention and whole cluster by pooling the attributes of individual mentions at the cluster level. In this way, the various cues known for the different mentions can be used simultaneously by the classifier or ranker, without any particular mention needing to be fully informative. We feel this formulation reflects cognitive

System	ACE 2002 -Sep02	ACE 2004	ACE 2005	CoNLL 2012	
				v5	v7
Luo et al. (2004)	89.9*				
Daumé III and Marcu (2005a)		89.2*			
Björkelund and Farkas (2012)				58.26	61.24
CherryPicker			63.4		
Stanford		56.6		59.23	54.21

Table 3.3: Performance of entity-level models on standard evaluation corpora.

\* indicates results are ACE scores using gold mentions.

theories of human discourse processing and note that its implementations have had some success in advancing the state of the art for coreference resolution.

### Combining Mention-Pair Scores

Luo et al. (2004) formulate coreference resolution as a search task over the set of possible partitionings or clusterings of mentions in a document. This being the case, the decision tree for how to process a document, mention by mention, is a tree with a Bell number (Bell, 1934) of leaves. The  $i^{th}$  layer of this tree represents the possible clustering states which can be reached by resolving the  $i^{th}$  mention to one of the existing clusters, or starting a new cluster; the correct resolution will be one of the exponentially increasing number of leaf nodes. Since exhaustive search over an exponential number of possible clusterings is intractable, inference proceeds by keeping track only of the best paths at each decision point. Luo et al. uses a capped maximum beam size of 20 paths, populating it with paths whose scores are within 0.1% of the best path. Additionally, only mention-cluster pairs which match in ACE entity type are considered.

Processing for each layer involves comparing the current mention against each predicted partial cluster (in each path in the beam), as well as scoring how likely the mention is to be discourse-new. On each mention-cluster comparison, the maximum of the pairwise comparisons is taken as the likelihood of the decision to resolve the current

mention to the given entity cluster. That is, Luo et al. selects the most informative mention in the cluster to inform inference, following the work of Ng and Cardie (2002b).

Reported results are based on using only mentions annotated in the ACE corpus, rather than having mention extraction as an automatic system component. When compared to their mention-pair baseline, this entity level model trails by 0.8% on the ACE 2002 Feb02 test set and 0.4% on the Sep02 dataset.

Daumé III and Marcu (2005a) generalise Luo et al.’s work by describing and testing various ways to formulate the function which combines mention-pair scores. Summing is ruled out since it could result in overcounting effects in large entity clusters. Average score, minimum score, and maximum score are suggested as mathematically meaningful. They additionally experiment using the last mention in the cluster, to emulate Soon et al. (2001) style mention-pair models. The best proposed scoring function is termed *intelligent* and uses a different combiner for each mention type. In particular, for name mentions, the score from the first name in the cluster is used, otherwise that from the last nominal. For nominal mentions, the maximum score from any nominal in the cluster is used, otherwise that of the closest name mention. For pronouns, the average of all scores from name and pronoun mentions is used. All combiners back off to the maximum pairwise score if the required mention types do not occur in a given cluster.

Simple combiners are all less effective, though vary in performance. Minimum score performs only 0.4% worse than intelligent, suggesting that it is possible to learn coreference by avoiding bad decisions, rather than promoting good decisions. Average score represents a drop of 1.0%, maximum score 2.5% and the score from the closest mention 5.2%. The poor performance of maximum score and closest mention explains the relatively poor performance of Luo et al. and *closest first* clustering, respectively.

Björkelund and Farkas (2012) placed second on the CoNLL-2012 shared task using a system which pipelined two coreference resolvers. The first was a mention-pair resolver, whose decisions were fed as features to the second, entity-level resolver.

The first, mention-pair classifier used closest first decoding for pronouns and best first decoding for non-pronouns since it was found that this yielded a 0.23% increase in CoNLL score above best-first clustering. The second, entity-level classifier adapted mention-pair scores by taking their geometric mean. When this entity-level classifier is used in isolation, it performs 1.2% worse on the CoNLL metric than the mention-pair baseline despite a 0.66% gain in CEAFE score (suggesting that it produces the correct *number* of clusters, though not finding all links within these). Pipelining it with their mention-pair classifier gave a 0.42% CoNLL score gain.

### Entity-Level Attributes

Yang et al. (2004) retain the binary classification formulation of coreference but introduce entity-level modelling by converting instances from being mention pairs to triples comprising the current mention, candidate antecedent cluster, and a representative mention from that cluster. While the reference mention is used to extract typical mention-pair features to inform the classifier, the inclusion of the candidate entity cluster is novel. It is used to extract globally-aware features which track the entity's linguistic attributes, surface form, and topicality based on those of the resolved mentions. In this way, the impact of some mentions being underspecified for these is minimised. For instance, the semantic class of a mention is typically known from NER processing, though that of nominals and pronouns is typically ambiguous.

The system is designed for the biomedical domain and evaluated on the GENIA<sup>5</sup> corpus, making its results not directly comparable to those of other systems reviewed here. We note, however, that incorporating entity-level features is shown to achieve a 2.8% F score increase, largely from boosting recall, which increases 2.6%.

Culotta et al. (2006) extend this work under the general task definition, proposing a rich set of entity-level features to encode insights targeted by mention-pair features at

---

<sup>5</sup><http://www.geniaproject.org/>

the level of emergent entities. Moreover, the mechanism for generating these features can be seen as a template for how to formulate entity-level features in the general case.

Binary valued mention-pair features become four-valued features depending on how many mention-pair comparisons return true: all-true, most-true, most-false, and all-false. As in Yang et al., cluster size and the linguistic attributes of type, gender, and number are important at the cluster level. The latter is captured in features which have the form all-X and most-X if all or a majority of the clustered mentions have been the same value for one of these linguistic attributes. Similarly, the authors use the output of a trained mention-pair classifier model over the mention pairs to define a four-valued feature according to the number of mention pairs predicted to be coreferential under this model.

Together with their contribution of how to define features at the entity level, Culotta et al. provide a novel inference algorithm and this requires a reformulation of how training instances are generated. In particular, inference involves initialising a set of singleton clusters, one for each of the extracted mentions, and repeatedly merging clusters until the model suggests no further merges. Culotta et al. opt to use a ranker rather than a classifier in their system since there may be two partially correct merges (in the case of split clusters) and the goal in this case should be to prefer the best merge, without penalising the less good, though still acceptable, merge.

To train this model, positive instances are generated by sampling a cluster from the gold standard and splitting it; the correct action is then to merge the fragments. Conversely, negative instances are generated by sampling two different annotated clusters, which should not be merged. Importantly, training instance sampling is error driven in that a sampled gold cluster is selected from those which would remain fragmented by the current model, and a sampled cluster pair would erroneously be merged. The performance of this system formulation is strong, improving  $B^3$  on an (undocumented) ACE dataset by 6.8%; this boost comes from boosting precision at the expense of recall.

Two publicly available systems develop from the work: Rahman and Ng (2009) extend the entity-level feature template in their ranking based system, CherryPicker<sup>6</sup>, while Stanford's (Raghunathan et al., 2010; Lee et al., 2011) multi-pass sieve system<sup>7</sup> is based on a similar inference algorithm, which incrementally merges partial clusters in several stages to produce coreference output.

CherryPicker processes documents by iterating over mentions in left-to-right reading order. For each mention, instances are generated comprising the current mention and one of the entity clusters which has thus far emerged. For training, instances are labelled for the ranker with preference 2 if they are positive in the gold standard and 1 otherwise. A special case is made for discourse-new mentions, which are modelled as coreferring with a dummy entity cluster representing this decision. All features used either pertain only to the current mention, or are four-valued translations of mention-pair features between the mention and the candidate antecedent cluster.

CherryPicker is not commonly used in research due to difficulties translating beyond the ACE task definition. In particular, it is designed for the ACE semantic classes rather than the unrestricted coreference task of OntoNotes and span mismatch drastically diminished observed performance since ACE only required the minimal span to be indicated in the output. The reported results (Table 3.4) are based on CoNLL evaluation in that they average MUC, B<sup>3</sup>, and CEAFE scores, though they are not comparable with other systems since they use their variation of B<sup>3</sup> (cf. Section 2.2.2) on an ACE corpus.

Stanford's system was the best performing system in the CoNLL-2011 shared task. It consists of a pipeline of deterministic *sieves* which take a set of clusters and attempt to merge compatible fragments. The sieve passes are arranged from high to low precision so that merges which happen early are more trusted, and the pooled attributes on the clusters are able to inform clustering for the low precision sieves. For instance, mentions which are aliases of one another are resolved before pronouns, so the gender of a person (from their given name) can be used to inform resolution involving gendered pronouns.

---

<sup>6</sup><http://www.hlt.utdallas.edu/~altaf/cherrypicker.html>

<sup>7</sup><http://nlp.stanford.edu/software/dcoref.shtml>

System	CoNLL 2012	
	v5	v7
Fernandes et al. (2012)	63.37	60.65
Björkelund and Kuhn (2014)		61.63
CL <sup>3</sup> M	63.30	60.00

Table 3.4: Performance of structured prediction models on standard evaluation corpora.

Due to its simplicity and integration with the CoreNLP<sup>8</sup> package, Stanford’s system has been heavily used in research, for instance in named entity linking (e.g. Hajishirzi et al., 2013) and slot filling (e.g. Angeli et al., 2013).

While these two systems develop from the same work, we see CherryPicker as remaining the most faithful to cognitive insights. In particular, it retains the natural human left-to-right reading order which means that discourse entities emerge as a document is processed in a way which perhaps mimics how mental space objects become related in the mental model of human readers.

### 3.1.5 Structured Prediction

Current state-of-the-art systems are based on structured prediction combining the strengths of mention-synchronous clustering and, to an extent, entity-level modelling. They work from the assumption that clusters emerge via mention-pair links which define a learnable structure over mentions in a document. This generalises the clustering strategies seen for the mention-pair model: the pairings selected for resolution should be those with the highest confidence among those possible for a document. At the same time, it is possible to capture some non-local information in models, and this is equivalent to defining entity-level attributes.

Fernandes et al. (2012) was the winning system for the CoNLL-2012 shared task. It uses tree structures to formulate coreference resolution as a structured prediction

<sup>8</sup><http://nlp.stanford.edu/software/corenlp.shtml>

problem. In particular, each node in a tree represents a mention and parent nodes are the antecedents of child nodes. The collection of entity clusters in a document is therefore a forest of trees, which is artificially cast to a tree by rooting each tree in a dummy parent. However, the authors note that they give no semantic interpretation to tree structure, which they describe as a by-product of structured prediction.

The weight of an edge connecting a mention-pair is taken to be the classifier confidence that the pair is coreferential. Inference proceeds by creating a graph of all possible mention pairings, assigning edge weights, and finding the graph's maximum spanning tree using the CLE algorithm (Chu and Liu, 1965; Edmonds, 1967). This means that inference is done over the document as a whole, rather than on a reading order pass over its mentions.

Björkelund and Kuhn (2014) extend Fernandes et al.'s structured model to incorporate non-local features which allow previous decisions to influence later ones. To achieve this, they decompose inference to iterate over mentions, on each iteration predicting the tree structure over the mentions so far seen, using best first decoding to select the candidate antecedent mention which is most compatible with the current mention.

To improve global consistency, beam search over candidate tree structures is used. Within this formulation, the authors find that system performance is highly dependent on the choice of beam search parameters and update strategy when no candidates in the beam are consistent with gold standard annotations. In particular, they only find performance gain over their local feature baseline Learning as Search Optimisation (LaSO, Daumé III and Marcu, 2005b).

In standard LaSO, when all predictions in the beam are incorrect, the standard perceptron update is made and the beam is reseeded from a correct analysis. Delayed LaSO uses the same search strategy but updates are retained in memory throughout document processing and applied after the whole document has been processed. Updates to the model are made using the passive-aggressive algorithm and a loss term

of 1.5 in the case of a false discourse-new prediction, else the hamming loss. Using standard LaSO with these parameters affords a CoNLL score increase of just under 2% and delayed LaSO just over 3% with respect to a greedy baseline.

Non-local features are included which encode the shape of the tree structure of the cluster containing the candidate antecedent, the cluster’s (minimum) distance from the start of the document, as well as a feature over the grammatical argument of clustered mentions. Including these features achieves a 1% CoNLL score gain, which at the time of publication, was the best reported performance on the CoNLL-2012 task.

Chang et al. (2013) similarly extend Fernandes et al. to include non-local features. As in Björkelund and Kuhn, inference is decomposed to iterate over mentions and incrementally produce a tree representation of the document’s clustering, though they do not use beam search to achieve non-local modelling. Instead they implement entity-level attributes as constraint terms that are added into the scoring function which is maximised in training. The terms which add to the scoring function promote the clustering if two mentions have (1) the same span length, or (2) same determiner plus a semantically related head word, or (3) are the same proper name. Conversely, the constraint terms which reduce a clustering’s score capture cases of incompatible modification, or incompatibility in the assignment of linguistic attributes: gender, number, professional title, or nationality.

Chang et al. acknowledge that the resulting objective function can be solved using ILP, but find that, since the constraints need to be given high weights, greedy inference produces similar results. While their results in Table 3.4 (styled as CL<sup>3</sup>M) are not as strong as those of Björkelund and Kuhn, their non-local features perhaps offer a valid alternative for how to incorporate entity-level attributes in structured prediction.

### 3.1.6 Summary

Based on this review, we propose to use an entity-level model in this thesis which directly models competition between candidate antecedent entity clusters. This model

is suggested both by cognitive theories of human discourse processing, and supported by promising results of systems which use entity-level attributes to inform extraction of non-local features. We want to retain natural left-to-right reading order when processing a document rather than mention-synchronous approaches which view the document as a whole since we consider these to be more faithful to linguistic models of the task.

While Björkelund and Kuhn (2014) produce promising results with structured prediction, we agree with Fernandes et al. (2012) that latent structure has no semantic interpretation. Furthermore, the increase in complexity of structured models comes at the expense of expressive power in that it becomes difficult to introduce entity-level features. Since we feel that the richness of entity-level features has produced some promising results, and is linguistically motivated, we opt not to pursue structured prediction.

## 3.2 Features

Another relevant facet of theoretical research in coreference explores how coreference is textually realised. By reviewing this literature, we now revisit our question from Section 3.1.1 of how identity of reference between mental space objects is realised textually between their linguistic expressions. We do this with the aim of understanding the current features informing computational models, as well as to identify shortcomings in these encodings.

In Section 3.2.1, we review the linguistic literature on reference, in particular anaphoric coreference. The breadth of this research highlights that coreference is a complex phenomenon straddling many domains including syntax, pragmatics, semantics, and discourse theory. Section 3.2.2 then explores how linguistic insights have informed feature development. We find that, while much of the strength of current state-of-the-art approaches is achieved through linguistically inspired features, there is

still a gap in the translation of linguistic theory, in particular Accessibility theory, to the computational setting.

### 3.2.1 Linguistic Description

Coreference and anaphora are often used in the NLP literature synonymously. However, in the linguistic literature, they are distinct: coreference is a semantic phenomenon in which two linguistic units denote the same entity, while anaphora is a dependence relationship between two linguistic units which may or may not indicate coreference. While this thesis uses the term *anaphora* broadly, we survey the linguistic literature on the dependence relation to better understand the nuances of coreference. We will see that there is no neat one-to-one mapping between coreference and anaphora. Instead, there are a range of linguistic cues which indicate possible coreference, whose summary in Accessibility theory grounds much of the work in this thesis.

#### Anaphora and Binding

Anaphora is a dependence relationship between two linguistic expressions in which one, the *anaphor*, is linked to another in its preceding context, the *antecedent*; this link from anaphor to antecedent is crucial for determining how the anaphor is interpreted by a reader. For instance, bridging anaphora grounds the referent of an anaphor. In ‘*Cole was hit by a bomb and the hole let water into the hull*’, ‘*the hull*’ anaphorically links to ‘*Cole*’ and thereby is interpreted as referring to ‘*the hull of Cole*’. That is, anaphora does not necessarily entail coreference. Also, coreference need not necessarily entail anaphora: two unambiguous names are coreferential if both refer to the same entity, regardless of their context.

However, in bound anaphora, anaphora and coreference do co-occur, as illustrated in the following except. The anaphor ‘*its*’ has an anaphoric link to the antecedent ‘*Cole*’ and by linking to it in this way, is interpreted to have the same referent.

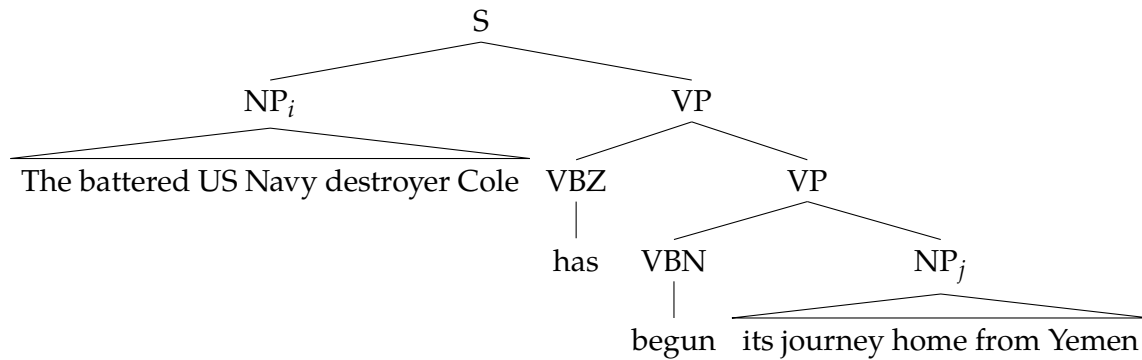


Figure 3.1: Example of c-command relationship in a constituency parse structure.

**The battered US Navy destroyer Cole** has begun **its** journey home from Yemen, 17 days after a suspected terrorist bomb tore a gaping hole in its side.

Bound anaphora is defined and studied in Government and Binding theory (Haegeman, 1991). The theory explains when coreference between anaphor and antecedent is certain (or impossible) using three principles based on the command structure of a sentence. An anaphor is said to be *bound* if it is *c-commanded* (constituent commanded) by a *coreferential* phrase. Figure 3.1 shows the c-command relationship:  $NP_i$  c-commands  $NP_j$  because the parent of  $NP_i$ ,  $S$ , spans  $NP_j$ .

The first principle is that anaphors which are coreferential with their antecedents must be bound. That is, since ‘*its*’ does not head its noun phrase and also is c-commanded by ‘*Cole*’, it follows that ‘*its*’ and ‘*Cole*’ are coreferential; if they were not, we would have a violation of this first principle.

The second and third principles state that an anaphor (pronouns in the second principle and other reference expressions in the third) which is not coreferential with its antecedent cannot be bound. For instance, the second principle dictates that ‘*him*’ cannot be coreferential with ‘*John*’ in ‘*John likes him*’ since ‘*him*’ heads its constituent and having ‘*John*’ and ‘*him*’ be coreferential would mean that ‘*John*’ binds ‘*him*’, since ‘*John*’ c-commands ‘*him*’. The third principle similarly precludes ‘*John*’ and ‘*his father*’ being coreferential in ‘*John likes his father*’.

Overall, Government and Binding theory formulates the common sense intuition that arguments of a verb (or other argument-taking unit) cannot be coreferential with one another, unless they are reflexive, in which case they must be coreferential. This makes sense since predicate-arguments structures typically predicate a relationship between two entities.

### Centering Theory

Centering theory (Grosz and Sidner, 1986; Grosz et al., 1995) extends the scope of Government and Binding theory, explaining how the antecedent of a (free) referential pronoun or full noun phrase can be found in the context preceding the expression. At its core, Centering theory describes the interdependence of coreference and discourse coherence. The linguistic expressions of interest are typically still pronouns, since coherence has “greater effect on the processing of pronominal expressions” than definite descriptions.

Centering theory is formulated around two levels of discourse structure and the entities which are salient therein. At the *global* level, a discourse as a whole is concerned with a particular topic and the entities, or centers, around which its narrative develops. At the *local* level, a discourse can be broken up into discourse segments, which are in turn comprised of utterances. Each utterance refers to a collection of entities, but, the authors claim, one most centrally. The collection of entities a utterance discusses is called its *forward looking centers*,  $C_f$ , and its central entity its *backward looking center*,  $C_b$ . Discourse coherence requires each utterance’s backward looking center being amongst the forward looking centers of a previous utterance.

As well as controlling discourse coherence, centering affects the choice of reference expression used in anaphoric links according to the rule:

If the backward looking centers of adjacent utterances are the same entity, pronominalisation of the center is strongly preferred. Else, a definite noun phrase is typically used, to mark the transition.

An example analysis is given in Figure 3.2. We divide the discourse into single-clause utterances, giving example entities and concepts which could reasonably be brought to a reader's mind as  $C_f$ . Importantly, the centers given in the figure do not correspond to coreference mentions: instead, they are abstract entities and concepts evoked by the textual mentions. The  $C_b$  of each utterance is indicated in bold.

The analysis shows that the discourse is largely coherent: '*USS Cole*' is the  $C_b$  of utterance 2 and is in  $C_f$  of utterance 1, and related entities bridge the following  $C_b$  to  $C_f$  transitions. Applying the pronominalisation rule, we can say that the definite noun phrase '*the ship*' is used in utterance 4 despite being pronominalised in the utterances 1 and 2, since it is not the  $C_b$  of utterance 3. However, utterance 5 represents a discontinuity: the huge Norwegian transport vessel is the most salient entity of the utterance ( $C_b$ ), but not among the  $C_f$  of utterance 4.

Grosz et al. (1995) formalise the ideas introduced by Grosz and Sidner (1986) and developed in the intervening years. Its key contribution is introducing terminology for understanding the different transitions in  $C_b$  and their impact on discourse coherence.

These rules rely on an extension of the model which claims that the  $C_f$  sets are partially ordered. This ordering measures the chance of a  $C_f$  being the  $C_b$  of the subsequent utterance according to Centering's proposed rule that it is the highest ranked  $C_f$  which is mentioned in the subsequent utterance which becomes its  $C_b$ . The authors state that a range of factors can impact this ordering, giving the examples of grammatical argument and syntactic parallelism: subjecthood and sharing a grammatical argument pushing a noun higher in the ordering. For instance, given that it is the subject of utterance 1, '*USS Cole*' is presumably its highest ranked  $C_f$  and indeed goes on to become the  $C_b$  of utterance 2.

The three transitions discussed in Grosz et al. are: center continuation, center retaining, and center shifting. Sequences of continuation are preferred over sequences of retention, which are in turn preferred over sequences of shifting.

- 1) The battered US Navy destroyer Cole has begun its journey home from Yemen,
- 2) 17 days after a suspected terrorist bomb tore a hole in **its** side.
- 3) **The attack** killed 17 American soldiers and wounded 39.
- 4) Flanked by warships and guarded by aircraft, **the ship** was towed out of Aden Harbor
- 5) to rendezvous with a huge Norwegian transport vessel
- 6) that will carry the crippled ship to the United States.

	$C_f$	$C_b$
1	USS Cole, United States, Yemen, Navy, journey	-
2	2000 bombing of USS Cole, bombs, terrorism, USS Cole	USS Cole
3	2000 bombing of USS Cole, United States, soldiers	2000 bombing of USS Cole
4	USS Cole, Aden Harbor, towing, warships, aircraft	USS Cole
5	Norwegian transport vessel, Norway, meetings	Norwegian transport vessel
6	USS Cole, United States	USS Cole

Figure 3.2: Centering analysis of the example excerpt.

In *center continuation*, the  $C_b$  of two adjacent utterances are the same and this center is the most highly ranked element in  $C_f$  of the second utterance. By having the second utterance's highest ranked  $C_f$  be the same center as its  $C_b$  means that it is the most likely center to become the  $C_b$  of the next utterance and discussion of its topic continues without interruption. This, for instance, would explain the observation that topical entities are repeatedly referred to with chains of pronouns.

In *center retention*, the  $C_b$  of adjacent utterances are again the same, but the center is not the most highly ranked in the second utterance's  $C_f$ . This transition signals a smooth transition of discourse topic away from the center. In *center shifting*, the  $C_b$  of adjacent utterances are not the same: the topic has changed, but not smoothly. Our analysis in Figure 3.2 is an example of center shifting, though the  $C_b$  of adjacent utterances, if not identical, are related.

### Accessibility Theory

Accessibility theory (Ariel, 2001) offers a more general formulation of the cognitive foundations of reference resolution. Its specific goal is to explain how speakers select what form a referential expression should have and how hearers use this cue when interpreting a referential expression in discourse. While Centering theory explained how speakers choose between pronouns and definite descriptions based on the salience of their related discourse entities, Accessibility theory has the broader scope of explaining the use of a diverse range of reference expressions, though salience is again the vital factor determining usage. Concretely, Accessibility theory builds from the notion that entities and concepts relate to human memory nodes which fluctuate in their degree of activation during a developing discourse.

The surface form used for a referring expression tells the hearer how accessible or activated the referent is, thus giving them an instruction about how to select between candidate entities and concepts as the referent. In Centering theory, pronominalisation was triggered when a center was highly ranked in salience among a set of centers; in

*Full name + modifier* < *Full name* < *Long definite description* < *Short definite description*  
 < *Last name* < *First name* < *Distal demonstrative + modifier*  
 < *Proximate demonstrative + modifier* < *Distal demonstrative + NP*  
 < *Proximate demonstrative + NP* < *Distal demonstrative* < *Proximate demonstrative*  
 < *Stressed pronoun + gesture* < *Stressed pronoun* < *Unstressed pronoun*  
 < *Cliticised pronoun* < *Verbal inflections* < *Zero*

Figure 3.3: Accessibility hierarchy of Ariel (2001).

Accessibility theory, pronominalisation instructs a hearer to retrieve a highly accessible entity, one corresponding to a highly activated memory node.

Accessibility theory contributes two key explanatory mechanisms. Firstly, it defines an accessibility hierarchy which arranges referring expression types according to the degree of accessibility they encode. This hierarchy is explained to result from three factors: *informativeness* (how much semantic content the expression contains), *rigidity* (how uniquely it refers to a particular entity), and *attenuation* (its phonological size, or how many phonemes comprise its pronunciation). For instance, pronouns are indicators of high accessibility since they have low informativeness, low rigidity, and high attenuation and full names are indicators of low accessibility since they have high informativeness, high rigidity, and low attenuation.

Secondly, it enumerates a range of factors which influence the accessibility of an entity or concept when seeking to resolve an anaphor mention:

- **Cohesion** The higher the semantic similarity between a discourse entity and the anaphor, the higher the accessibility of the entity.
- **Proximity** The shorter the distance between a mention of a discourse entity and the anaphor, the higher the accessibility of the entity.
- **Parallelism** Clauses which are more cohesively linked entail more dependency in their interpretation. For instance, rephrasings or extrapolations of the description

of an event are likely to refer to the same entities, with parallel syntactic and semantic structures. This, for instance, would explain why entities tend to persist in subject position if they are the agent in a narrative.

- **Topicality** Entities which are global discourse topics are more accessible than those which are local discourse topics, which are more accessible than entities which are not topical.
- **Competition** If multiple compatible entities compete for the role of being the anaphor's antecedent, each is less accessible than it otherwise would be.
- **Automaticity** of the inference required to resolve the entity impacts its accessibility in that cases where resolution requires complex inference have diminished accessibility presumably since these create high cognitive load for the hearer.

These factors are found to correlate with determinations of accessibility in corpus analysis, but no one factor is sufficient by itself. Additionally, Ariel notes that at times, factors can conspire and indicate the same accessibility value. For instance, topics tend to be mentioned more frequently than non-topics and this frequency alone means that mentions are likely to be close to one another. Drawing on a range of factors not only gives the model rich expressive power, but we also see its framework to be highly compatible with machine learning in that Accessibility theory and machine learning both weigh information across a multi-dimensional space in order to determine a classification. However, explanations are different from generating a prediction, and no algorithm is given for doing this.

There is a growing body of psychological work demonstrating that how humans resolve pronoun reference is impacted by factors such as those listed in Accessibility theory (e.g. topicality and inference in Kertz et al., 2006; Rohde et al., 2007; Kehler and Rohde, 2013). The work in this thesis is complementary to this effort in that the two provide alternative ways to assess the usefulness of cognitive theories: psychological experiments assess how well the theories describe human discourse processing while

our work assesses how well they can be applied to automation. We note also that this psychological work splits reference form (specifically pronoun) understanding into two sub-tasks, production and resolution. In such models, the above listed factors are split according to whether they influence our production or resolution of reference expressions in language: grammatical and salience-based factors influence production, while semantic and pragmatic inference-based factors influencing resolution. The work in this thesis accords with such models in that we argue that cohesion-based features are insufficient for modelling coreference resolution and that encoding more complex inference is essential for improving current performance.

In the next section, we will see how Binding and Centering theories, as well as the Accessibility hierarchy and the theory’s enumeration of factors impacting entity salience, have been encoded as features in computational models. A key goal of this thesis is to address gaps in this effort, and, so, we draw heavily on these theories for motivating our feature development work.

### **3.2.2 Feature Review**

We now survey the features used in the systems discussed in Section 3.1, using the factors used in Accessibility theory to categorise the review. We do not review competition here since it has been covered at the model level in competition learning approaches. We aim to study a feature based on where it was first proposed because the trend is that all subsequent systems will use all features described in earlier work.

The feature set for coreference resolution is now quite large and diverse, representing the aim of the CoNLL shared tasks to promote research in linguistically rich modelling. Despite this, we find that there is still a substantial amount of insight encoded in linguistic theory which has yet to be applied to the computational task. This thesis represents an effort to identify this gap, and the best way to address it.

## Binding

Ng and Cardie (2002b) incorporate three features which capture the intuitions from Government and Binding theory. In particular, the first indicates whether a c-command relationship exists between the pair of mentions being compared, based on the given constituency parse structure. A separate feature looks for whether two mentions have the same maximal noun phrase, presumably to capture nouns which take arguments and are thereby subject to binding constraints. In our example in Figure 3.1, ‘Yemen’ and ‘its journey’ both have the same maximal noun phrase ‘its journey home from Yemen’ and their coreference is ruled out by this feature. The third is an alternative implementation of the first two, using span overlap to approximate c-command.

Raghunathan et al. (2010) move to using dependency parse arcs in place of tradition constituency parse trees. They forbid any clusters forming that will contain mentions headed by tokens related to the same verb via the subject (*‘nsubj’*), direct object (*‘dobj’*), or indirect object (*‘iobj’*) relationship.

## Accessibility Hierarchy

Accessibility theory gives no mechanism for applying its hierarchy, but all modern coreference systems use its typology of mentions to some extent. In particular, models classify mentions into three and five broad categories: proper name, definite, indefinite and demonstrative nominal, and pronoun.

One way to use such information is to encode features to capture how likely a mention of a particular type is to be included in a coreference relationship, and which transitions between mention types are allowed. Of the five features which build from the Accessibility Hierarchy of Soon et al. (2001), only two are incorporated in their final decision tree for MUC-6. These two encode whether the current mention is a pronoun and whether the candidate antecedent is a pronoun: pronouns are highly likely to be anaphoric, thereby involved in coreference. The three features which are not included indicate whether the current mention is definite noun phrase, is a

demonstrative noun phrase, and whether both the current mention and candidate antecedents are both proper names: a majority of nominals are not anaphoric and more sophisticated reasoning than these features is required to determine whether or not to include them in coreference clusters.

Ng and Cardie (2002b) implement similar features to Soon et al., but also include four extra features to indicate whether both mentions are definite noun phrases, or both are pronouns, and whether the candidate antecedent is a definite noun phrase, or is an indefinite noun phrase. We note that a “both mentions are pronouns” feature accords with Centering theory since it is an example of the most preferable transition, center continuation. Stoyanov et al. (2010b) extend this still further, by introducing a  $4^2 = 16$  valued feature to reflect whether the mentions in the pair are proper name, definite nominal, indefinite nominal, or pronoun.

An alternative way to use this information is to specialise modelling by mention type. In this way, different weights can be learned for a given indicator, such as proximity, for the different mention types. For instance, pronouns are more likely to be close to their antecedents than other mention types are. Denis and Baldridge (2008) learn five different models, one for each of the mention types: proper name, definite nominal, indefinite nominal, third person pronoun, and non-third person pronoun. This specialisation achieves an increase in CoNLL score of 1.1%, with CEAFE increasing the most (1.6%). The latter result suggests that specialisation enables the model to choose the correct number of entity clusters.

Bengtson and Roth (2008) and Durrett and Klein (2013) implement specialisation at the level of features within a model, rather than explicitly learning separate models. Bengtson and Roth prefix each base feature generated with the type of the current mention, one of proper name, nominal, or pronoun. On the other hand, Durrett and Klein use the large training dataset of OntoNotes to learn a model over three versions of each base feature: unprefixed, conjoined with the type of the current mention, and conjoined with concatenation of the types of the current mention and candidate

antecedent mention. Additionally, a finer-grained categorisation is used: the type of a pronoun is given by its normalised form (e.g. *'his'*, *'him'*, and *'he'* share the same mention type, *'he'*). While the impact of specialising features in this way is not explicitly tested, these two systems are highly competitive with similar mention-pair frameworks, suggesting that intuitions from the Accessibility Hierarchy are particularly suited to coreference resolution.

## Cohesion

Cohesion is a primary target in the development of features for modern coreference resolution systems. Semantic similarity is a fuzzy concept, and has been targeted at various levels including mentions' surface forms, morphosyntactic attributes, and their lexical semantics. Lexical semantics requires knowledge external to a document, and so has been a difficult factor to encode. The overall trend we see is that, as indicators of semantic cohesion broaden to cover fuzzier relationships, improvements in system performance plateau. We suggest that one explanation for why these features have yet to realise a substantial performance gain is that their increase in descriptive power requires stronger, more discriminative models to correctly rank the candidates they generate.

**Surface Form** The first surface form feature explored was string match: if the surface form of two mentions are the same, they are likely to refer to the same discourse entity. Despite its simplicity, it is a very strong feature and remains a vital feature in current state-of-the-art systems.

In Soon et al. (2001), string match indicates whether the surface form of two mentions were the same up to their articles and demonstrative pronouns. Their decision tree using just this feature covers 66.3% true positive examples in MUC-6 and achieves 53.9% MUC score on MUC-6 and 54.3% on MUC-7. Expanding from string match, Soon et al. uses an alias feature to check whether two mentions have compatible, if not identical,

surface form. In particular, they do comparisons between a normalised form of date expressions, the last word of person names, and acronyms of organisation names. A decision tree using just this alias feature covers 11.5% true positive examples in MUC-6 and achieves 57.7% precision, but only 3.9% recall.

Soon et al.'s string match features are both over- and under-productive. The authors' error analysis over a sample of five test documents from the MUC-6 corpus reveals that 71% of spurious coreference links are due to mentions having the same surface form but different referent, while 63% missing links cannot be resolved by their surface form match heuristics.

String match has remained little changed since these first features, except for adjustments to suit the mention annotations of a given corpus. We have seen how Ng and Cardie (2002b) boosted their system's precision compared to their Soon et al. baseline by decomposing into different features for different mention types. Raghunathan et al. (2010) and Björkelund and Farkas (2012) adapt the string match heuristic to the CoNLL shared task mention annotations, Raghunathan et al. using it to indicate whether two mentions had the same surface form up to possessive markers and Björkelund and Farkas up to determiners, possessive markers, and punctuation.

To address the under-productivity of string match, Ng and Cardie (2002b) introduce a range of features which look at sub-string match and do matching over the words in mentions. Concretely, they encode features which indicate whether there is a sub-string match between the surface form of two mentions, or whether there are any content words in common between pairs. To capture the role of modification in restricting reference, they encode a feature which indicates whether the pre-nominal modifier words of one mention is a subset of those of the other. Unfortunately, these features decreased performance by increasing recall at the expense of precision, particularly on nominal mentions. They therefore were not included in the final system reported in Ng and Cardie. Similar features, however, are vital to the strong performance of the Stanford system (Raghunathan et al., 2010). This system, for instance, restricts

coreference between any mentions where a later mention introduces a modifier not in a previous mention.

An alternative solution to boosting the productivity of surface form match is head match: mentions which have the same head word are likely to be coreferential since the head word of a phrase captures its core semantic meaning. For instance, the head of both phrases ‘*the crippled ship*’ and ‘*the ship*’ are the same, ‘*ship*’; the modifier ‘*crippled*’ is extra information. Culotta et al. (2006) introduce three head match features. The first checks simple head match while the second indicates whether head words are substrings of one another and the third accounts for rephrasings by indicating whether the pre-nominal modifiers of one mention match the head or the pre-nominal modifiers of the other. All current state-of-the-art systems use head match to inform inference.

Other inexact match features proposed include edit distance between mentions, normalised by mention length (Denis and Baldridge, 2008; Stoyanov et al., 2010a,b), and whether the length of the two mentions is the same (Bengtson and Roth, 2008).

**Morphosyntactics** One layer deeper than surface level features are morphosyntactic features which are derived by rule-based processing of a mention’s surface form. The morphosyntactic attributes commonly used are gender, number, animacy, and person; with some exceptions, mentions need to agree in these attributes to be coreferential. These attributes are derived from the POS and NER tags of a mention’s head word, gazetteer lookup of gendered first names, and known properties of pronouns.

Agreement in gender and number was introduced by Soon et al. (2001). Ng and Cardie (2002b) introduce agreement in animacy, as well as a conjoined feature encoding agreement in both gender and number. Feature conjunctions are expanded in later work, with Stoyanov et al. (2010a,b) using features to capture whether both mentions are pronouns and agree in gender, number, and person, and Culotta et al. (2006) generalising the idea by producing the Cartesian product of their attribute match features. To model when attributes are compatible, if not identical, Denis and Baldridge

(2008) use sparse feature values which are pairs of attribute values from the paired mentions.

**WordNet** (Miller, 1995) is a lexical semantic resource which organises the senses of a word form as objects called synsets, and arranges these synsets into an ontology according to their semantic relationships (e.g. hypernymy). One aim of the resource is to capture common noun usage, making it a potentially valuable resource for understanding relationships between nominal mentions in a document. However, WordNet features are typically not included in modern systems given their small impact on performance.

We outline here two key ways it has been used to inform coreference resolution systems. In the first, it is used to assign coarse-grained semantic class to nominal mentions which mimic the NER annotations on proper name mentions. In this way, semantic class can be used in linguistic attribute features akin to those just discussed: agreement in semantic class is an indicator for coreference between mentions. Soon et al. (2001) do this by mapping the NER labels of proper name mentions to synsets and mining all hyponyms of these synsets to produce a gazetteer for each label. In this way, if a nominal mention is headed by a word in one of these gazetteers, it gets the relevant class label and semantic class match can be used to link it to a name in its context. Soon et al.'s semantic class feature was not included in their final decision tree because the assignments were found to be very noisy and the categorisation too coarse-grained.

Bengtson and Roth (2008) use this same methodology as Soon et al. to assign semantic classes, but translate it to the ACE categories. Additionally, they use a generalised notion of attribute compatibility in defining feature values to be pairs of coarse grained semantic classes from the two mentions.

The second way that WordNet is used to model the cohesion between mentions exploits the ontology structure of the resource directly. These features are generated by mapping a mention to a synset via its head word and encoding the relationship between

these synsets. Ng and Cardie (2002b) encode features which indicate whether a hypernym path exists between the synsets of the mention-pair and, if so, another with the length of this path. This second feature is based on the approximation that path length tracks semantic similarity. However, like their substring and word match features, these WordNet features boost recall at the expense of precision and are dropped for their final system. On the other hand, Culotta et al. (2006) use the fact that synonymous words are mapped to the same synset, introducing features indicating synonymy and antonymy. These features are taken up by the competitive Reconcile and UIUC systems, the former of which includes a feature whose value is the first WordNet synset that both mentions share as a feature, presumably as an attempt to learn which regions of WordNet can be trusted.

Since first sense is a good baseline for word sense disambiguation, as well as to minimise processing time, it is typical to map a mention to a synset via the first sense of its head word. Ponzetto and Strube (2006) argue that doing so fails to address semantic ambiguity and introduces sense proliferation. They introduce a series of features based on a range of available semantic similarity metrics (Rada et al., 1989; Wu and Palmer, 1994; Resnik, 1995; Jiang and Conrath, 1997; Leacock and Chodorow, 1998; Lin, 1998b). In particular, feature extraction calculates the similarity between each synset pair in the Cartesian product of all synsets for the mentions head words; feature values are then the maximum or average similarity score from these calculations. The best performing WordNet uses the maximum Wu and Palmer similarity, while introducing all similarity features improves the performance of a weak baseline by 6.3% and 2.2% on ACE 2003 BNEWS and NWIRE datasets.

**External Knowledge Sources** The general consensus is that deeper knowledge than surface-level and morphosyntactic features is required to model semantic cohesion between mentions, though we have seen that WordNet has been unable to address this. In particular, WordNet has scattered coverage of proper names, making it unsuitable

for encoding why proper name and nominal mentions are semantically related to one another. There is an active body of research on how to use the information in encyclopaedic knowledge bases for coreference resolution. In particular, Ponzetto and Strube (2006) explore using Wikipedia, Uryupina et al. (2011) and Rahman and Ng (2011) using YAGO (Suchanek et al., 2007), and Hajishirzi et al. (2013) using Freebase.

Additionally, researchers have sought to mine encyclopaedic knowledge from unlabelled text, based on patterns which are high-precision indicators of coreference. Hearst (1992) describes a general method for discovering these patterns and a study of how reliably six such patterns find relationships annotated in WordNet. The largest pattern finding study for coreference resolution is described by Haghighi and Klein (2009), who use bootstrapping to detect patterns in BLIPP and Wikipedia to mine pairs of proper names and nominal descriptors. These patterns boost coverage of recall errors on coreference from ‘little more than half’ to 67%.

### **Proximity**

The preference for coreferential mentions to be close to one another is used in all approaches surveyed in this chapter. All features are formulated based on the assumption that documents are single discourse units without segmentation. Modelling any segmentation in documents would mean modifying the following features to indicate enhanced distance in the case of a segment breaks, such as across sections in a report or shifts in topic of a conversation.

Approaches vary in how they quantify distance in a document. Soon et al. (2001) count the number of sentences between the mentions; their decision tree classifier learns a rule on whether mentions are in the same sentence. By extension, Ng and Cardie (2002b) measure distance with the number of paragraphs between mentions, since paragraph structure is given for the MUC corpora. Combining these features, Stoyanov et al. (2010a,b) include features which indicate whether mention pairs are in the same sentence, or in the same paragraph.

Ng and Cardie introduce a feature encoding whether a candidate antecedent is the closest compatible mention to the current mention since this is likely to be the correct resolution. This is taken up in the rule-based Stanford system (Raghunathan et al., 2010), which deterministically resolves a pronoun to its closest compatible antecedent. Bengtson and Roth (2008) generalise the idea, measuring mention distance to be the number of compatible mentions between a mention-pair, allowing for distractor mentions.

Both Fernandes et al. (2012) and Durrett and Klein (2013) use number of mentions to measure distance, but do not check for compatibility. Instead, distance from a pronoun or person name mention counts the number of proper name mentions, indicating the number of entities mentioned in the context. In their state-of-the-art approach, Björkelund and Kuhn (2014) propose a novel distance based feature: the distance (in number of mentions) of a candidate entity cluster from the start of the document. This measurement presumably addresses entity clusters fragmentation, by learning to disprefer late emergence of fragments.

### **Parallelism and Topicality**

Parallelism is the linguistic tendency for similar mentions to surface in similar contexts. Parallelism reduces the load on a hearer in that they can use similarity of context as a cue for similarity of reference. Similarity of context has been encoded at the level of a mention's grammatical argument (e.g. whether the mention is a subject or object of a phrasal verb), and its semantic role (from FrameNet, Baker et al., 1998).

Topicality, on the other hand, relates to the salience of an entity in its discourse. Topical entities are mentioned frequently, in prominent contexts. Therefore, features over grammatical arguments also capture topicality.

**Grammatical Arguments** When considering the grammatical argument of mentions, parallelism says that mentions should take the same grammatical argument in adjacent

sentences. To this end, Ng and Cardie (2002b) encode two features which capture whether both mentions are subjects or are embedded as modifiers of larger noun phrases, respectively. Additionally, four single-mention features capture whether the current mention or the candidate antecedent, individually, are in the subject position, or embedded.

Ng and Cardie's features can also be seen to relate to topicality in that topical entities prefer strong contexts, notably the subject position. So, these features capture the tendency for parallelism and topicality to conspire in giving the same indication of accessibility: topical entities tend to repeatedly appear in the subject position and, thereby, make good targets for coreference.

Ponzetto and Strube (2006) expand Ng and Cardie by defining sparse features whose values are a pair with the first element being the grammatical argument of a mention and the second element the predicate governing the mention. These features do not directly capture parallelism in that comparisons are not made between feature values of the two mentions, though they provide a finer-grained model of the topicality of the mention in the sentence. They yield a 4.9% gain on BNEWS in ACE 2003.

Björkelund and Kuhn (2014) learn transitions in grammatical argument with sparse features. In their entity-level model, they define features whose values are pairs where the first element is the grammatical argument of the current mention and the second is that of the candidate antecedent. Such features are able to capture whether two mentions are both subjects, but also the likelihood that two objects, or non-parallel arguments, are coreferential.

**External Knowledge Sources** FrameNet has been used as a source of external knowledge for frame semantic parallelism and will be reviewed in Chapter 7.

## Inference

Inference is the capability of humans to extrapolate from incomplete pieces of information to deduce likely conclusions. In the case of coreference, humans are able to use contextual cues to decide the likely resolution of a mention, even when this resolution is not necessarily indicated by the other cues we have seen. We review how systems have encoded inference based on the relationship of speakers and pronouns in their quoted speech, as well as how approaches to the Narrative Cloze task model consistencies in narrative structure and referential patterns.

**Quoted Speech** Culotta et al. (2006) include a feature which captures whether two mentions are both the attributed speaker of quotes, to capture the assumption that cohesive quoted text has come from the same speaker. More directly, Stoyanov et al. (2010a,b) and Raghunathan et al. (2010) include heuristics to identify mention pairs with the relationship that one mention is the pronoun ‘I’ (*‘we’*) and in a quoted span whose attributed speaker is the other mention, which has been NER tagged as person (organisation).

**External Knowledge Sources** Systems on the Narrative Cloze task (Chambers and Jurafsky, 2009) have been used to generate datasets for frame semantic inference. They will be discussed in Chapter 7.

## Lexicalised Features

The features reviewed in the previous sections are hand-engineered features based on linguistic insight. Orthogonal to this are recent efforts which use highly sparse, lexicalised features to approximate linguistic information.

Fernandes et al. (2012) introduce two feature classes comprising features whose values are the surface form (POS tag) of a mention’s head word, previous two words, and next two words. Both the current mention and the candidate antecedent are used

to extract features. Fernandes et al. won the CoNLL-2012 shared task despite not using entity-level modelling, suggesting the power of these lexicalised features.

Durrett and Klein (2013) similarly employ lexicalised features in their mention-pair model to achieve competitive performance. Their lexical features have values which are the surface form (with backoff to POS tag in the case of infrequent words) of a mention's head word, first word, last word, previous word, and next word, as well as the length of the mention. Again, both the current mention and the candidate antecedent are used in feature extraction. Durrett and Klein find that these lexical features perform similarly to hand-engineered, linguistically-motivated features in their system, and argue that this is because they target the same information, albeit implicitly. For instance, the first word feature having value '*the*' is equivalent to having a hand-engineered feature capturing whether a mention is a definite noun phrase.

### 3.2.3 Summary

Our review has shown a great diversity in the features explored to model coreference resolution computationally and that, whether implicitly as in the case of hand-engineered features or explicitly in the case of lexicalised features, these are motivated by linguistic models of reference resolution. Some explanatory mechanisms of Accessibility theory, particularly lexical cohesion, have been heavily explored, while for others, we can see gaps in the literature. In particular, our current typology of mentions as being proper names, nominals, and pronouns is too coarse-grained given the Accessibility Hierarchy; incorporating inference based on frame semantics has yet to yield substantial improvement despite its intuitive appeal; and how to model competition in a way that is aware of the relative salience of the competing entities is not at all obvious. These questions frame the following chapters of this thesis.

### 3.3 Summary

We have reviewed the literature on reference resolution, using the insights in the linguistic and cognitive literature to frame our review of the computational literature. We have seen that coreference resolution has most profitably been modelled by entity-level approaches informed by a diverse range of features.

We ground the remainder of this thesis on this understanding. Chapter 4 develops our cognitively inspired system, *LIMERIC*, which is an incremental, entity-level approach to coreference resolution, which achieves performance competitive with the state of the art despite its simple design. Chapters 5, 6, and 7 expand the space of features available to inform our model, competing with the state of the art by drawing on insights from Accessibility theory.

## 4 Incremental Coreference Resolution

*Work described in this chapter forms part of the conference paper Kellie Webster and James R Curran. 2014. Limited memory incremental coreference resolution. In Proceedings of the 25th International Conference on Computational Linguistics, pages 2129–2139.*

In this chapter, we describe LIMERIC, the low memory, incremental coreference resolution engine we design to capitalise on insights discussed in the last chapter. We start by drawing an analogy between shift-reduce parsing and coreference resolution to reformulate resolution as a series of shift and reduce operations which incrementally produces a collection of entity clusters as a document is read in a single left-to-right pass. By designing the entity collection to be self-ordering, it comes to have cognitive meaning, being a simplified model of the human mind. Furthermore, LIMERIC naturally incorporates the strengths of both non-local decoding and entity-level modelling for achieving globally consistent decisions.

We then identify our requirements for system design, before detailing our Python implementation. We use the features reviewed in the last chapter as our baseline feature set and explore its strengths in an ablation study and via feature weights. We find that LIMERIC is very good at learning discourse information and that semantic cohesion features have decreasing impact as they broaden to capture fuzzier relationships. We benchmark LIMERIC and find that, while it is simpler and has lower memory requirements, it is competitive with the best approaches to the task, achieving 64.22%

and 59.99% CoNLL F score on the 2012 shared task benchmark using gold and automatic preprocessing, respectively.

## 4.1 Motivation

In designing LIMERIC, we draw an analogy between coreference resolution and shift-reduce parsing (Section 4.1.1), and use Centering and Accessibility theories to motivate implementing the entity cluster forest as a self-ordering list (Section 4.1.2). By incorporating these insights into the task, we develop a model which naturally incorporates entity-level modelling into a best-first clustering framework (Section 4.1.3), and is highly flexible for the development of rich linguistic features we develop in the next chapters. It also enables us to define coreference resolution and anaphoricity determination (the task of classifying mentions as discourse-new or anaphoric) as a joint task (Section 4.1.4).

### 4.1.1 Shift-Reduce Parsing

The shift-reduce algorithm (Aho and Johnson, 1974) is widely used to parse text in languages (both programming and natural) due to its efficiency, simplicity, and its low memory requirements. It takes as input a *sentence* (string of symbols) and outputs either a parse tree representing the syntactic structure of the sentence, or a value indicating that the sentence is not valid in the language. For programming languages, symbols correspond to units such as keywords, literals, and operators; for natural languages, tokenised words.

The algorithm is designated  $LR(k)$  to denote that it processes a sentence from left ( $L$ ) to right, reading  $k$  symbols at a time. During processing, the syntactic parse is incrementally generated as each symbol is read, as shown in Figure 4.1. Intermediary parse units (sub-trees of the final parse tree) are stored and operated on in a collection denoted the *forest*. At each step, the parser performs one of two operations, namely

Forest	Operation	Queue
	Initialise	The President said he and his wife ...
DT   The	Shift	<u>The</u> President said he and his wife ...
DT    NNP        The President	Shift	<u>President</u> said he and his wife ...
NP /   \ DT   NNP       The President	Reduce	said he and his wife ...
NP            VBD /   \          DT   NNP    said       The President	Shift	<u>he</u> and his wife ...

Figure 4.1: Series of shift and reduce operations creating a syntactic parse tree.

*shift* or *reduce*. In a shift operation, the input symbol is removed from the sentence and a new tree representing just this symbol is added to the right frontier of the forest. For instance, the first shift operation in our example removes ‘*The*’ from the sentence and creates its DT-labelled tree to the forest. In a reduce operation, a new node is created and this becomes the direct ancestor for a number of trees at the right frontier of the forest. In the reduce operation in our example, we can see that a new noun phrase (NP) node was created which spans ‘*The president*’ since there is an English grammar rule which says that an NP can be composed of a determiner followed by a noun. Only nodes at the right frontier are candidates for reduce operations.

Since both operations take place solely at the right frontier of the forest, it can be implemented using a push-back stack.

When the shift-reduce algorithm is  $LR(1)$  only one symbol is read from the sentence at a time. In such a framework, a sentence is always read from left to right without look ahead or backtrack. Therefore,  $LR(1)$  can be implemented with a queue whose

elements are symbols in the language. This is the typical implementation of shift-reduce parsing, though such a model is limited to deterministic context-free grammars and, therefore, can not fully model natural languages. Despite this limitation, we take  $LR(1)$  as the basis for developing our novel algorithm for coreference resolution with the goal of exploring how well it is able to model the task. With this as the baseline, future work could consider if introducing a  $k$ -mention look-ahead affords substantial gains. In our cognitive interpretation of our model, look-ahead would model the ability of humans to scan ahead when resolving reference, which is supported by eye-tracking experiences (for a review, see Rayner, 1998).

To develop our algorithm, we draw an analogy between how sentences are processed in parsing and how we would like to process sequences of mentions in coreference resolution. In particular, we see mentions as the symbols of our reference ‘language’ and their coreference relationships as the structure we would like to predict. As such, the queue data structure should store the mentions extracted from a document; that this is read exactly once without look ahead is linguistically meaningful since this is what a human reader of a document does. Similarly, the forest should store the entity clusters which emerge incrementally as a document is processed. In this way, a document may be processed with a series of two operations: mentions can either shift into the forest if they are the first mention of a new discourse entity, or reduce with an emerging cluster if it corresponds to its discourse entity. These operations are represented in Figure 4.2.

With this analogy established, we now need to decide how much of the shift-reduce specification can be applied directly to coreference, and which aspects require reformulation. Firstly, where it makes sense for elements of the forest to be trees in syntactic parsing, the desired output of coreference resolution is clusters of mentions, which need not have internal structure. While there have been models which do propose internal tree structure for coreference clusters (Fernandes et al., 2012; Björkelund and Kuhn, 2014), we do not feel it is necessary; furthermore it is not at all clear whether

Forest	Operation	Queue
	Initialise	<i>'The President', 'he', 'his wife', 'his' ...</i>
[ <i>'The President'</i> ]	Shift	<i>'<u>The President</u>', 'he', 'his wife', 'his' ...</i>
[ <i>'The President'</i> ], [ <i>'he'</i> ]	Shift	<i>'he', 'his wife', 'his' ...</i>
[ <i>'The President', 'he'</i> ]	Reduce	<i>'his wife', 'his' ...</i>
[ <i>'The President', 'he'</i> ], [ <i>'his wife'</i> ]	Shift	<i>'<u>his wife</u>', 'he' ...</i>

Figure 4.2: Series of shift and reduce operations creating a collection of emerging entity clusters.

such structure is linguistically meaningful. The elements of the forest in our system therefore contain lists of mentions, extended with linguistic attributes and other entity level information we will define.

A key difference between the structure of syntactic and anaphoric relationships is that the latter cannot be written as a context-free grammar. This is problematic since it is the grammar rules of a language which dictate what reduce operations will take place. We instead formulate this as a machine learning problem, and define a classifier to fill this role in Section 4.2. Lastly, unlike syntactic rules, coreference relationships are not projective: where syntactically related elements appear close to one another in a sentence, entities increase and decrease in salience throughout a discourse. Therefore, we cannot limit our attention to the rightmost frontier of the forest and this cannot be implemented with a push-back stack. To decide how to implement the forest, we consider theories of cognitive science.

### 4.1.2 Cognitive Insight

Both Centering (Grosz et al., 1995) and Accessibility (Ariel, 2001) theories model human cognitive processing with a simple collection of entities which are stored and tracked as a discourse develops. We have seen that both theories associate each entity in this container-type store with a degree of salience relative to the other contained entities. This degree of salience is inherently non-uniform and dynamic: at any given point in a

discourse, entities differ from one another in their degree of salience, and the salience of a given entity increases and decreases as it comes in and out of topic.

In the local-level of Centering theory, each utterance within a discourse segment is associated with a list of entities, of centers, about which the utterance concerns. These lists are, in later formulations of the theory, ordered by the salience of each entity and they in turn populate a stack-like store; the most recent entity list is pushed onto the stack, and related to other lists on the stack via anaphoric links.

We therefore propose that the forest data structure from the shift-reduce algorithm should reflect the salience of the entities it contains. The data structure we propose is a self-ordering list. In this way, accessible entities will tend to be found to the right of the list, repeatedly promoted if they are central to the discourse. In the same way, incidental entities will drop away from the right frontier as the discourse progresses without referencing them further. This develops from the entity store described in Centering theory, by flattening the list of lists structure into a single list. It also makes concrete the idea that entities are more accessible if they are salient.

In Accessibility theory, it is not specified how entities are stored, though its description of how their relative salience is rated is more detailed than that of Centering theory. Among the many factors which impact accessibility, a key one is proximity: entities which have been mentioned recently are more salient than entities which have not been. We therefore order the forest data structure by recency as an approximation of salience. While we expect the correct ordering to take into account topicality and other factors, we only use depth in the forest of discourse entities via coarse-grained bucketing and relative positioning. In this way, we expect any small inaccuracies in entity ordering to have minimal impact on the models we derive.

### 4.1.3 Entity-Centric Design

The analogy between shift-reduce parsing and coreference resolution offers us a neat way to incorporate entity-level modelling into a best-first clustering framework. Entity-

level modelling is straightforward: the self-ordering entity list contains incrementally growing entity clusters similar to the discourse entity objects described in Fauconnier’s (1994) Mental Space theory and Recasens et al.’s (2011) development of the theory in their near-identity proposal.

We incorporate non-local decoding by requiring that, in the case of a reduce operation, classification select the *best* target cluster among the candidates in the forest, rather than the rightmost compatible entity cluster. The implication of this is that we must search the entire forest to determine the correct target for any reduce operation, not just the right frontier as was the case for syntactic parsing. While enforcing a full search gives our process worst case  $O(n^2)$  time complexity in the number of mentions, this worst case only occurs in incoherent document in which each entity cluster contains exactly one mention. We anticipate this not to occur in real world data, particularly OntoNotes data, where entity clusters have average size around four mentions. In the average case, exhaustive forest search still represents a time saving compared to full mention-pair models which compare each mention against all candidate antecedents.

#### 4.1.4 Anaphoricity Determination

Anaphoricity determination is the task of classifying mentions according to whether they are coreferential with a previous mention. It is related to, but distinct from, coreference resolution which requires systems to decide which candidate is the correct antecedent of an anaphoric mention. In our framework, it is the task of classifying whether the next move should be a shift or a reduce.

Ng and Cardie (2002a) presented a supervised approach to anaphoricity determination which used features similar to those used for coreference resolution, including a mention’s type, whether it had a head match with any other mentions, as well its position in the document. While the system achieved 86.1% and 84.0% accuracy on MUC-6 and MUC-7, incorporating its decisions into their coreference system as a filter on which mentions an antecedent would be sought resulted in a drop in performance

by 0.1% and 3.2% on the datasets. The authors found that this drop was from recall and resulted from the anaphoricity classifier erroneously labelling anaphoric mentions non-anaphoric. By bypassing the anaphoricity classifier on mentions which are aliases of or have a string match with another mention in the document, anaphoricity determination realised a performance gain, of 2.0% and 2.6% on MUC-6 and MUC-7.

Ng (2004) expanded this work, identifying two dimensions which influence the impact of anaphoricity determination on coreference resolution. Specifically, the output of an anaphoricity classifier can either be included in a coreference system as a constraint (as above), or as an extra feature to the coreference classifier. Also, anaphoricity determination can be a modular component, optimised independently of coreference, or can be jointly learned along with coreference. He finds that the jointly-optimised resolver using as a constraint outperforms the other three possible frameworks, improving performance on ACE by up to 4.5%, 3.2%, and 2.8% on broadcast news, newspaper, and newswire. The most informative features are found in analysis to be among those used by Ng and Cardie (2002a): head match, string match, and mention type. More recently, CherryPicker (Rahman and Ng, 2009) finds that joint anaphoricity determination outperforms a pipelined filter by 0.6%, 2.2%, and 2.9% MUC, B<sup>3</sup>, and CEAFE on ACE 2005.

Related to anaphoricity determination for the OntoNotes corpus is the task of singleton detection. Singleton detection requires a system to predict whether a mention is coreferential with any other mention in the document, not just whether it has a backward looking reference to a previous mention. Given that singleton mentions are not annotated in OntoNotes, singleton detection is the task of labelling mentions as reportable or not. Recasens et al. (2013) analyses how certain linguistic properties of mentions impact the mention's likelihood of being a singleton and finds that the strongest indicators of singletohood are inanimacy, indefiniteness, quantification, and a high degree of modification. A classifier trained on the observed tendencies achieves

an F score of 80.7% on the task of identifying singletons in OntoNotes, and affects an improvement of 0.47% CoNLL score on CoNLL-2012.

Durrett and Klein (2013) jointly learn coreference and singleton detection by implementing a high recall mention extraction component and relying on relevant features to be down-weighted if a given mention is not reportable. This approach has the advantage of being agnostic as to whether singletons are annotated in the input data, and is reasonable given the similarity between anaphoricity and coreference features sets, and the reliance of important anaphoricity features (i.e. head and string match) on coreference-like comparisons. We therefore employ it in our work below, though extend the ideas in Recasens et al. in our lifespan score features (cf. Section 4.3.1) and revisit the problem of how best to model anaphoricity determination and singleton detection in Chapter 6.

## 4.2 System Design

Following this motivation, we implement our system around two key data structures, a queue of mentions and a self-ordered list of entity clusters. Initialising our system involves extracting mentions from a document and populating the queue and is described in Section 4.2.1. We learn a classifier which is jointly trained to decide whether the next operation should be a shift or a reduce and, if a reduce, which cluster the mention should merge with. Inference is described in Section 4.2.2 and training in Section 4.2.3.

### 4.2.1 Initialisation

For our system to be comparable with the current state of the art, mention extraction needs to be designed to suit the OntoNotes guidelines (Pradhan et al., 2012). We ignore verbs since they represent a small proportion of annotated mentions, and we expect their linguistic behaviour to be substantially different from that of nouns. We follow

Durrett and Klein (2013) and aim to extract a mention for each noun phrase annotated in a document. By maximising recall in this way we learn a model that is robust to spurious extraction in preprocessing: missing an extraction labelled in the gold standard will always yield a recall penalty, but a classifier can be designed to learn that certain mentions should not be reported.

To populate the queue, extracted mentions are sorted into top-to-bottom, left-to-right reading order: mentions with spans starting earlier are ordered first; in the case of mentions starting on the same token, longer mentions precede shorter ones. OntoNotes guidelines stipulate that in the case where candidate mentions share the same head word, the candidate with the longest span is annotated. To implement this, we simply search for noun phrases (nodes labelled NP or NML) by traversing the provided constituency parse trees from their root, extracting a mention from the first indicated node seen with a given head. Since noun phrase annotations are typically flat, we also extract as mentions any tokens POS tagged as pronouns, as well as any token sequences labelled as entity names by named entity (NE) annotations, if their span is not already seen as a noun phrase, or only differs from an extracted mention by a known honorific or possessive marker. Entity name extractions exclude QUANTITY, CARDINAL, and PERCENT spans, following (Raghunathan et al., 2010).

The mentions we extract will not have a perfect mapping with the mentions annotated in the gold standard. We use a three-stage back-off processing to align the two sets. We started development using just two stages: first, we aligned any (gold, extracted) pair where the mentions have the same span, then we aligned any remaining (gold, extracted) pair where the mentions have the same head subject to the constraint that head-matched pairs contain the same number of conjuncts. The number of aligned extracted mentions using these two stages, as well as the number of missed gold mentions and spurious system mentions, are given in Table 4.1.

We can see that, as expected, the proportion of missed mentions is indeed small, especially for the dataset with gold standard annotations for the preprocessing layers of

	Aligned	Missed	Spurious
Gold TRAIN	152294	3266	230438
Gold DEV	18733	423	29279
Auto TRAIN	149963	5597	234301
Auto DEV	18361	795	29846

Table 4.1: Number of mentions extracted from the TRAIN and DEV portions of OntoNotes 5, using two-stage mention extraction.

part-of-speech, named entities, and parse structure (cf. Section 2.1.3). Also, the number of spurious extractions is very high. We anticipate that the large number of spurious mentions will impact the profile of our system, but not the quality of its output: having more mentions to process will increase runtime, and introduce substantial bias into the training sample, but training can be designed to be robust to this signal.

Categorising the 423 mentions missed in processing Gold DEV, 30% (127) are single token verbs while the remainder form a long tail of assorted problem cases. On automatically preprocessed documents, there is almost twice as many reportedly missed mentions. While we expect the number of missed mentions to be higher than for gold preprocessed documents due to noise, a substantial amount of this gap is due to mentions being sub-optimally aligned. We therefore introduced a third stage in which remaining (gold, extracted) pairs are aligned if all of the following conditions are met. The statistics for the three-stage back-off process are given in Table 4.2

- neither have length 1; and
- both spans start at the same token; and
- the extracted mention covers at least half the tokens of the gold mention; and
- the extracted mention is the candidate which maximises overlap with the gold mention.

While the statistics do not change substantially for documents with gold preprocessing, the number of missed mentions for automatically preprocessed documents has greatly reduced. Surveying the cases which remain unable to be aligned, we find

	Aligned	Missed	Spurious
Gold TRAIN	152349	3211	230383
Gold DEV	18742	414	29270
Auto TRAIN	150965	4595	233299
Auto DEV	18517	639	29690

Table 4.2: Number of mentions extracted from the TRAIN and DEV portions of OntoNotes 5, using three-stage mention extraction.

that most correspond to single token verb mentions, parse errors (no phrasal unit exists from which the mention could be extracted), and named entity recognition errors (no named entity span exists from which a mention could be extracted).

### 4.2.2 Inference

The inference algorithm used to achieve our shift-reduce inspired processing of a document is given in Algorithm 1. By design, this algorithm is simple since we wish to avoid the complex algorithms applied by others to the task in favour of a more intuitive, linguistically motivated solution.

```

initialise queue;
initialise forest;
for mention : queue do
    prediction = classify(mention, forest);
    cluster = apply(mention, forest, prediction);
    promote(cluster, forest);
end
report();

```

**Algorithm 1:** LIMERIC’s inference algorithm.

To process each document, we read the enqueued mentions exactly once, in reading order without look ahead, and execute three key steps: classify, apply, and promote. The predicted entity clusters are then prepared for output, in report.

**classify** The task our classifier is given is to decide whether the next operation should be a shift or a reduce operation; in the case of it being a reduce operation, the classifier additionally needs to output which is the best candidate entity cluster among those in the forest. We choose an averaged perceptron classifier (Collins, 2002) due to its successful application to the task (e.g. Bengtson and Roth, 2008; Stoyanov et al., 2010a,b, 2011).

To do the classification, we generate a score for the likelihood of the next operation being shift and scores for each of the possible reduce operations, one for each cluster in the forest. Since the discourse properties of first mentions is qualitatively different than that of subsequent, anaphoric mentions, we do this using separate weights vectors for the two operations:

$$score_{shift} = f_{shift} \cdot \phi_{shift}$$

$$score_{reduce_i} = f_{reduce_i} \cdot \phi_{reduce}$$

Where  $f_{shift}$  and  $f_{reduce_i}$  are the feature vectors extracted for the shift and a given reduce operation, and  $\phi_{shift}$  and  $\phi_{reduce}$  are the independently maintained weight vectors for the two operations. Features are generated on-the-fly to reduce memory requirements, and because the state of the system is determined by each move made. All features are binary valued. Shift is selected if  $score_{shift}$  is greater than the  $score_{reduce}$ , where  $score_{reduce}$  is the maximum candidate reduce score. If reduce is selected, the target cluster is the one which maximises  $score_{reduce_i}$ :

$$target_{reduce} = \underset{i}{\operatorname{argmax}}(score_{reduce_i})$$

Since we are using our classifier to decide between a shift and reduce operations, it is learning anaphoricity; since we are also using it to decide between candidate targets for reduce operations, it is learning coreferentiality. That is, we jointly learn anaphoricity and coreferentiality, rather than pipelining the two processes.

**apply** In apply, the current mention joins the proposed entity cluster, or starts a new one. To achieve entity-level modelling, we want the cluster to have attributes which reflect all mentions it contained to facilitate making globally consistent decisions. That is, we want membership in an entity cluster to, in itself, be meaningful: a cluster may contain a mention which, in isolation, is underspecified with respect to some attribute but another for which that information is known. For instance, a mention like *‘it’* cannot be assigned a coarse-grained semantic class but, if it is in a cluster with *‘The battered US Navy destroyer Cole’*, we can use this clustering to know that it refers to a PRODUCT. To achieve entity level attributes, we pool the following properties among the clustered mentions:

- animacy
- gender
- grammatical number
- coarse-grained semantic class
- if a pronoun, its normalised<sup>1</sup> form
- lifespan score
- text of all tokens in a mention span
- text of all premodifier tokens in a mention span
- text of the head of a mention span
- if a conjunction, the number of conjuncts

---

<sup>1</sup>nominative

**promote** To ensure that the forest is self-ordering by recency, the target of a reduce operation needs to be removed from its position in the forest and moved to the right frontier. This is a crucial implementation detail given the cognitive interpretation we give to the forest of clusters. However, it is potentially a costly step: using a standard Python list means that the cluster must be first located before it may be removed, giving the operation  $O(n)$  in the number of seen entities. To counter this, we implement a custom doubly-linked list, giving this step constant time complexity.

**report** The generated entity clusters are postprocessed according to OntoNotes annotation guidelines. Specifically, singleton clusters are removed, along with clusters containing only bare plural mentions and those containing an indefinite nominal after the first mention.

### 4.2.3 Training

```

initialise queue;
initialise stack;
for  $i : n\_iterations$  do
    for  $mention : queue$  do
        prediction = classify(mention, forest);
        gold = correct_classification(mention, forest);
        if  $prediction \neq gold$  then
            update(prediction, gold);
        end
        cluster = apply(mention, forest, gold);
        promote(cluster, forest);
    end
end

```

**Algorithm 2:** LIMERIC's learning algorithm.

The learning algorithm we use to train our classifier is given in Algorithm 2. Again, we see that processing a document involves reading the enqueued mentions exactly once, in reading order without look ahead. As each mention comes to head the queue, we generate a training instance in which the classifier decides whether it is more likely that the mention shift into the forest as the first mention of a new discourse entity or reduce with the cluster of an already active one. If the classification is incorrect, the relevant weight vectors are updated toward the correct classification.

Although spans for gold mentions are available in training, we opt to train on automatically extracted mentions to match the conditions as far as possible between training and testing. This is especially important given the alignment statistics we observed in Table 4.2.

**classify** The classification procedure follows as described for inference, with only one point of difference. After Fernandes et al. (2012), we implement a large-margin interpretation of the perceptron algorithm. The aim of a large-margin classifier is to increase the margin of separation between positive and negative training instances. We achieve this by augmenting the scores of all non-gold classifications by a set amount so that any prediction has to win by at least this amount to satisfy the no-update condition. In our experiments we set this margin parameter to be 1.

**correct classification** We read the correct classification for an (automatically extracted) mention from its alignment with a gold mention. If the mention has not been aligned, it is a spurious extraction and the correct decision is to for it to shift into the forest, where it can remain a singleton cluster for later filtering.

On the other hand, if the mention has been aligned, the correct classification is found by looking at the cluster to which the gold mention belongs. In particular, if the aligned gold mention is cluster initial (Figure 4.3a), the correct classification is shift. Otherwise, the correct classification is to reduce with the entity cluster in the forest containing mentions aligned to the same gold cluster. For instance, in Figure 4.3b,

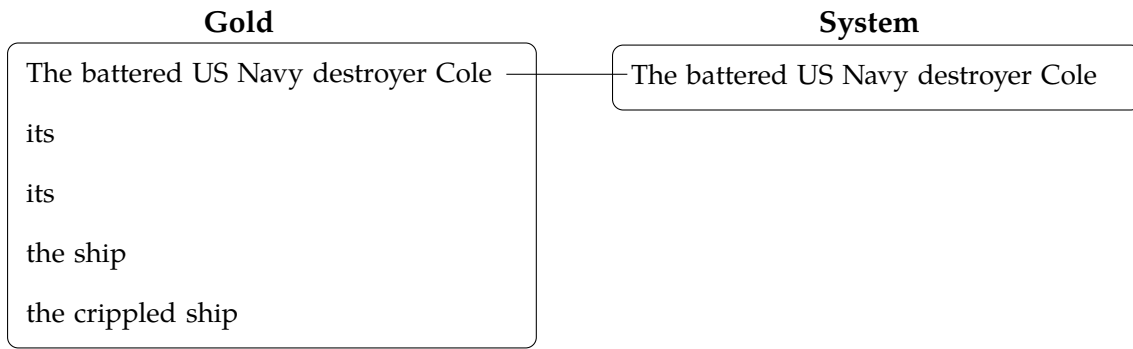
the correct classification is for *'the ship'* to reduce with the cluster [*'The battered US Navy destroyer Cole', 'its', 'its'*]. If there is no such cluster (see Figure 4.3c), the correct classification is shift since this corresponds to the case of a missed mention in automatic extraction. Since spurious extractions will never be a reduce target (since they have no gold links), their entity clusters will remain singletons. By learning that spurious mentions should remain singleton clusters, we develop a system in which we jointly learn singleton detection with coreference resolution.

**update** If the classifier mis-classifies the instance, we update the weight vectors toward the correct classification. That is, we increase the weights of all features corresponding to the gold classification, and decrease the weights of all features corresponding to the incorrect prediction. For instance, if a mention triggers an incorrect shift prediction, weights in  $\phi_{shift}$  for that mention will be decreased while weights in  $\phi_{reduce}$  for the comparison with the correct target entity will be increased. In order to balance the impact of our negatively-biased training sample, we do not adjust the weights corresponding to any other comparisons.

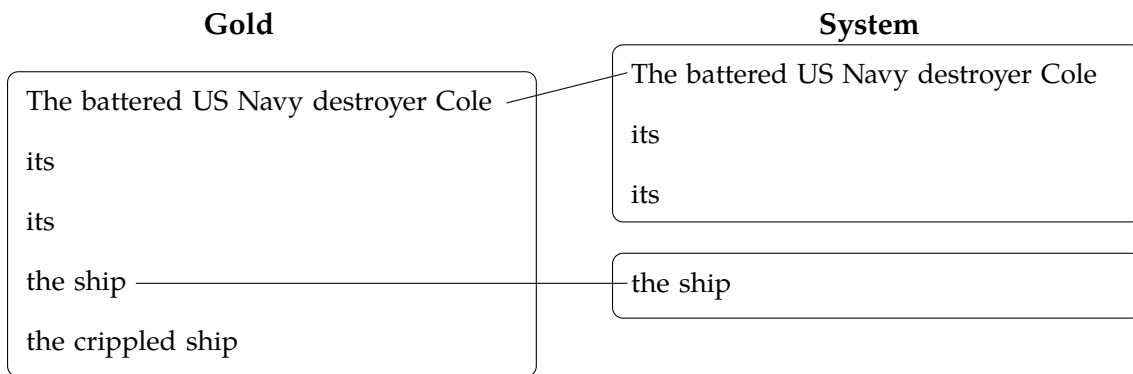
In a standard perceptron, the change in feature weights is uniformly 1; we make two adjustments to this. A common adjustment to standard perceptron updates is to use the Margin Infused Relaxed Algorithm (MIRA; Crammer and Singer, 2003) to determine the update value. The aim of this algorithm is to determine the minimum update values which are needed to bring the perceptron to a state where it will correctly classify the instance. By updating the perceptron using MIRA, classifiers are less prone to oscillate between bad states by applying too large an update, or converging too slowly on a good state by applying too small an update.

We implement MIRA updates using the following to determine the update value,  $\delta$ :

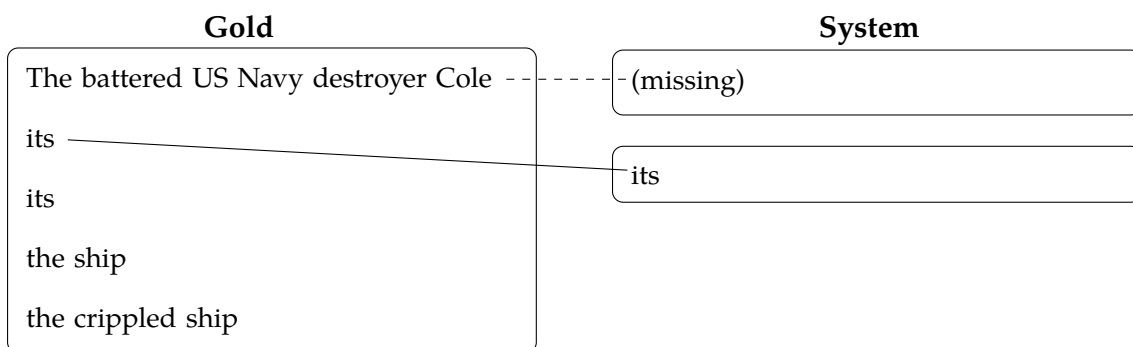
$$\delta = \frac{margin}{|P| + |G|}$$



(a) When an aligned gold mention is cluster-initial, the correct classification is shift.



(b) When an aligned gold mention is not cluster-initial, the correct reduce target is read from mentions in the same gold cluster.



(c) When an aligned gold mention is not cluster-initial, but there is no previous aligned gold mention, the correct classification is shift.

Figure 4.3: Determining the correct classification from gold standard annotations.

Where *margin* is the difference between the winning score and that of the correct classification,  $|P|$  is the number of features activated by the prediction, and  $|G|$  is the number of features activated by the correct comparison.

As a result of how feature sets are typically defined for coreference resolution, many more features are generated on a reduce comparison than on a shift comparison. During development, we noticed that this difference was negatively impacting performance by making reduce operations unfairly favourable. To grow the shift feature weights faster, we introduced a scaling parameter on the update of these feature weights; we found the ratio of the feature space sizes to work well. In particular, the value used to update the weights of shift features is scaled up by the ratio of the (larger) number of reduce features, divided by the (smaller) number of shift features, as represented in Algorithm 3.

```

 $\delta_+ = \text{margin} / (|P| + |G|);$ 
 $\delta_- = \text{margin} / (|P| + |G|);$ 
if predict shift; gold reduce then
    |  $\delta_- *= |G| / |P|;$ 
end

if predict reduce; gold shift then
    |  $\delta_+ *= |P| / |G|;$ 
end

```

**Algorithm 3:** Algorithm for determining feature weight update value.

**apply** As the final stage, the system applies the decided move. In training, there are two valid ‘decided’ moves, namely the correct transition, read from the gold standard, or the (potentially incorrect) predicted classification. In this work, we train by following the path of correct transitions, since we found the signal to be too noisy when following predicted classifications.

## 4.3 Features

As we saw in Chapter 3, feature development for coreference resolution has produced a diverse range of features; in order to establish a baseline for our next feature development chapters, as well as to understand the strengths and weaknesses of our current modelling, we implement an extensive set of these features (Table 4.3). We find that LIMERIC is very good at learning discourse patterns, and that the impact of cohesion features decreases as they broaden to capture fuzzier relationships.

### 4.3.1 Implementation

The impact of the mention-pair model on the evolution of coreference resolution has resulted in the majority of documented features being factored over two mentions. In using an entity-level model, we need to adapt these features for our system and we do so by identifying a number of general-purpose feature transformations, described next. We expand the set of objects over which features may be factored by including a novel depth feature which is defined with respect to the current state of the forest. We also introduce lifespan score as a novel way to incorporate Recasens et al.’s (2013) insights into a joint framework.

#### Feature Transformations

We outline four strategies for transforming features factored over mention-pairs into features which compare entity clusters (candidate antecedents) with a mention (the anaphor).

**cluster level attributes** Since pooling attributes means that entity clusters have a collection of properties previously only stored at the mention level, we are able to apply mention-pair features using these attributes with minimal change. In particular, we can use the same feature functions, only with a revised signature.

Class	Description
Grammar	argument number of the anaphor verb governing the anaphor mentions have a common NP ancestor mentions have a subject-object relationship mentions have a span overlap
Surface Cohesion	mentions have an exact string match mentions have a relaxed string match (Björkelund and Farkas, 2012) mentions have a substring match mentions have a head match mentions have a head substring match edit distance between mentions' head words mentions are an acronym and its expansion number of words in common number of premodifier words in common anaphor introduces new proper name modifier mentions have the same length
Attribute Cohesion	mentions agree on animacy mentions agree on gender mentions agree on coarse-grained semantic class mentions agree on grammatical number mentions disagree on pronoun normalised form anaphor is a conjunction number of conjuncts in anaphor mentions disagree on number of conjuncts anaphor length, in tokens
Lexical Cohesion	Lin (1998a) similarity of mentions' head words mentions are synonymous in WordNet mentions' first shared sense in WordNet
Proximity	distance between mentions, in number of sentences distance between mentions, in number of mentions depth of antecedent, counting all entity clusters depth of antecedent, excluding singleton entity clusters depth of antecedent, counting only NE entity clusters
Discourse / Topicality	anaphor is indefinite nominal antecedent size, in number of mentions pairing's lifespan score (feature prefixing)
Lexicalised	text (POS, shape) of anaphor's (closest antecedent mention's) head text (POS, shape) of the first token of the anaphor (closest mention) text (POS, shape) of the last token of the anaphor (closest mention) text (POS, shape) of the token directly preceding the anaphor (closest mention) text (POS, shape) of the token directly following the anaphor (closest mention)

Table 4.3: Baseline feature set of LIMERIC.

Where it does not make sense for attributes to be expressed at the cluster-level, we evaluate features pairwise against each mention in a cluster. We then determine the feature value from the results of these pairwise comparisons.

**any** Return True if any pairwise comparison returns True. This transformation should maximise the observed compatibility of pairs, particularly where an indicator is expected to be sparsely attested.

**best** Use the most compatible pairwise comparison to determine the feature value.

**count** Return the number of pairwise comparisons which returned True. This feature transformation is useful for strengthening the compatibility between mention-cluster without yielding multiple features per positive comparison.

### Grammar

Grammar comprises three traditionally mention-pair features, which are implemented with the **any** transformation: a syntactic violation with any clustered mention should count against the entity cluster as a whole. The features capturing whether mentions share a common NP ancestor are defined using the constituency parse structures included with the OntoNotes, and subject-object relationship using the predicate-argument annotations.

### Cohesion

We define cohesion features over three levels of information, the surface form of mentions, their linguistic attributes, and the lexical semantic relationships of their head words. Values for the linguistic attributes of animacy, gender, and grammatical number are assigned heuristically using similar strategies to Raghunathan et al. (2010). Coarse-grained semantic class is also defined using Raghunathan et al.'s heuristics for pronouns, and Soon et al.'s (2001) WordNet method for nominals. Lexical semantic

relationships are determined using the first sense of a nominal head word, since this is typically a good baseline for word sense disambiguation.

**Surface Form Cohesion** As well as strict string match in which no normalisation is performed, we incorporate the relaxed string match formulated by Björkelund and Farkas (2012), in which punctuation and possessive markers are ignored. Björkelund and Kuhn additionally ignored determiners, but we found better performance including them in comparison strings.

Features over the number of words and premodifiers in common are defined over the pooled word and premodifier lists to maximise the model’s ability to identify compatibility from sharing many words, or incompatibility from restrictive modification.

Features over the exact match of mentions’ string and head words are transformed using the **count** transform, with a maximum value of 5, while those over relaxed forms of matching use the **any** transformation. The one exception is head edit distance, which uses the **best** transformation (with feature value capped at 5) since we would like to return the edit distance of the most similar head words involved.

**Attribute Cohesion** We implement traditional features capturing the cohesion of features based on animacy, gender, semantic class, and number. As a specialised variant of grammatical number agreement, we include features over the number of conjunctions in mentions, determined heuristically from the parse structure of a mention. We include these features since we found that mentions with more conjuncts were less likely to participate in coreference. All are implemented with cluster-level attributes.

**Lexical Semantic Cohesion** There exist a range of metrics to gauge lexical semantic similarity (cf. Ponzetto and Strube, 2006). Ng and Cardie’s (2002b) similarity feature uses path length in the WordNet ontology, which is compromised from WordNet granularity being not consistent throughout. We instead use (Lin, 1998a) similarity, formulated with the **best** transformation such that the clustered mention which is most

related to the active mention is used in feature generation. Similarity values are binned  $[0.0, 0.2]$ ,  $(0.2, 0.6]$ ,  $(0.6, 0.8]$ ,  $(0.8, 1.0]$ .

### Proximity

We include two complementary ways to measure proximity.

**distance** Implementation of the common distance metrics, transformed using the **best** transformation to give the distance between the mention and the closest mention in the cluster. Distance is measured in two different features, and capped at 10.

Since depth from the right frontier of the forest in our model represents relative cognitive accessibility, we introduce **depth** features as the cognitive analogues of **distance**.

**depth** Index with respect to the right frontier of the forest. Since this is inherently a cluster level feature, no transformation function is required. We do, however, bin the values; the bins we define represent the depths  $\text{top} = [0]$ ,  $\text{upper} = [1, 2, 3, 4, 5]$ ,  $\text{lower} = [6, 7, 8, 9]$ ,  $\text{bottom} = [10, \dots, \infty]$ .

We define three variants of the depth feature, each designed to filter incidental discourse entities which may not have decayed from the accessible portion of the forest but are nonetheless likely to be outside the attention of the reader.

### Discourse / Topicality

**cluster size** The number of mentions in a cluster is expected to reflect the topicality of its referent, with large clusters corresponding to topical entities.

**lifespan score** We introduce **lifespan score** to model whether a mention is expected to remain in a singleton cluster. Lifespan score is a numeric feature based on the regression co-efficients presented by Recasens et al. (2013). To calculate lifespan features, each

mention is assigned a score which is the sum of the regression coefficients for the singleton indicators it satisfies. An alternative formulation could simply base each mention's score on the probability value given by Recasens et al.'s regression classifier; however, we opted not to implement scoring in this way to allow for indicators to be easily dropped from or added to future work.

By assigning mentions lifespan scores, we would like to learn that mentions with a high score should remain as singleton clusters and mentions with low scores should merge and form larger coreference clusters. When classifying a 'shift' operation, the value of the lifespan feature is the value of the active mention's score; when classifying a 'reduce' operation, it is the sum of this with the lifespan scores of the mentions in the candidate antecedent cluster. In both cases, lifespan score is binned by flooring the resulting lifespan score from floating point to integer value. In this way, the feature will disprefer large clusters, particularly those containing mentions which should remain as singletons. We found that summing performed better than averaging lifespan scores (which would counter the effect of cluster size on lifespan score) and attributed this to averaging 'blurring out' differences captured in the scores.

**Feature Prefixing** After Durrett and Klein (2013), we generate 'prefixed' features: multiple variants of each feature generated. Each time a feature from Table 4.3 is generated, we activate three distinct features. The first is unadorned, the second is specialised by the type of the current mention, and, for reduce operations, the third is specialised for the discourse transition being proposed. Concretely, if a feature  $X$  is activated on a reduce comparison between the mention 'he' and the cluster ['The President'], we would generate the three features  $\langle X \rangle$ ,  $\langle \text{mention=pronoun}, X \rangle$ , and  $\langle \text{name} \rightarrow \text{pronoun}, X \rangle$ , since 'he' has type pronoun and the transition from the last mention in the cluster to the current mention is from proper name to pronoun.

While prefixing inflates the size of our feature set, the features generated are more meaningful since we would expect many indicator functions to behave differently on,

for example, a pronoun anaphorically referring to a mention in the same sentence, and a proper name reintroducing an entity mentioned several sentences ago. Also, since we use perceptron learning, feature weights are only tuned if the feature is useful in making a decision during training.

We note that prefixing is not a feature type itself, but include it under the class of discourse features since the prefix labels capture discourse transition patterns.

### Lexicalised

We implement the data-driven features explored by Fernandes et al. (2012) and Durrett and Klein (2013). For each of the following three variants, we include five distinct feature types for the tokens mentioned in Table 4.3. The resulting fifteen feature types are generated both for the anaphor mention as well as the closest antecedent mention in the candidate entity cluster.

**text** The surface form of the given token. Since these features are, by design, sparse, Durrett and Klein uses a frequency threshold on their generation. They found that only generating text features for tokens which were seen at least 20 times in the training data worked well in their system. We find a threshold of 50 works better for our system.

**POS tag** Durrett and Klein (2013) uses a back off to POS tags in the case where a lexical item does not meet their lexical frequency threshold. We include this as an independent feature.

**word shape** To generalise patterns in morphology, proper names etc. we define analogous features to **text** and **POS tags** with the shape of the word, where shape reflects the evaluation of:

- Is the token text allcaps? Title case? All lower case?
- Is the token text numeric?
- Does the token text contain a hyphen character?

	MUC	B <sup>3</sup>	CEAFE	CoNLL
<b>LIMERIC</b>	73.24	61.29	59.07	64.53
- Grammar	73.04	60.99	58.56	64.20
- Surface Cohesion	69.92	56.64	54.81	60.46
- Attribute Cohesion	71.64	59.32	57.42	62.79
- Lexical Cohesion	73.06	60.97	58.72	64.25
- Proximity	65.98	53.45	50.14	56.52
- Discourse / Topicality	66.83	52.17	48.72	55.91
- Lexicalised	69.40	57.20	53.80	60.13

Table 4.4: Ablation analysis over CoNLL-2012 DEV using gold preprocessing.

- Does the token text end with a known suffix<sup>2</sup>?

### 4.3.2 Analysis

We study our feature set in an ablation study as well as in feature weight analysis. To understand the operation of our system, as well as to validate our novel depth features, we profile the forest of discourse entities.

#### Ablation

Table 4.4 gives the performance of each model in our ablation study on the CoNLL-2012 DEV portion of OntoNotes using gold preprocessing. We first run LIMERIC with the complete feature set described above, then remove features according to the classes given in Table 4.3. Each model is retrained with CoNLL-2012 TRAIN.

The feature classes with the largest impact on system performance are proximity and discourse / topicality, whose removal degrades the CoNLL score by over 8%. Furthermore, discourse / topicality, along with lexicalised, is also the class which accounts for the largest number of features via our prefixing strategy. That is, the performance of LIMERIC is strongly tied to its large feature set.

<sup>2</sup>-ed, -ing, -ion, -er, -est, -ly, -ity, and morphological -s

Considering the impact of removing discourse / topicality and lexicalised features, we can see that CEAFE is the most sensitive of our metrics to these changes. Moreover, these drops in CEAFE stem from a larger drop in recall than in precision (11.54% and 6.27% compared to 8.81% and 4.01%). Given the algorithm used to calculate CEAFE (cf. Section 2.2.3), we can infer that these features stop our model from missing gold entity clusters: cohesion features, the major focus of coreference research, are not yet sufficient for making all decisions of coreference, especially in cases where there are discourse cues such as proximity to inform the decision. Indeed, Accessibility theory makes the stronger claim that cohesion features are inherently insufficient for resolving reference and that all mechanisms it identifies are required.

Among the cohesion features, surface form cohesion is more important for system performance than attribute cohesion which, in turn, is more important than lexical semantic cohesion. This is consistent with the trend we noted in the last chapter whereby the performance of systems plateaus as features broaden to include fuzzier relationships.

### **Feature Weight Analysis**

Since we use a linear model, it is possible to analyse feature weights to introspect system performance. We do this by reporting the number of unique features in the complete LIMERIC model above with non-zero feature weight and their average magnitude. We use the same feature classes described previously, and additionally subclass features from surface cohesion, proximity, and lexicalised in order to understand the diversity in these broad feature classes.

Exact surface form cohesion captures exact string and head match, as well as mention length match; measured includes continuous-valued features: head edit distance and number of words or premodifiers shared; relaxed takes in the remainder of the surface form cohesion features, various forms of inexact match. We can see that the relaxed subclass has the highest average feature weight of the three subclasses, as well

Class	Features	Weight
Grammar	1746	0.119
Surface Cohesion	2713	0.131
exact	885	0.115
relaxed	808	0.148
measured	1020	0.130
Attribute Cohesion	8697	0.134
Lexical Cohesion	3326	0.068
Proximity	3363	0.113
distance	2093	0.110
depth	1270	0.117
Discourse / Topicality	4964	0.047
Lexicalised	371489	0.067
text	334497	0.063
POS tag	23246	0.104
word shape	13746	0.109

Figure 4.4: Number of distinct features and their average weight in LIMERIC, by feature class and subclass.

as the fewest distinct features: it compactly provides a good model of surface form cohesion. Comparing relaxed surface form, attribute, and lexical cohesion, we can see that the average feature weight decreases from one to the next. This is again consistent with our observation that performance plateaus as cohesion features broaden to capture fuzzier relationships.

The two strongest feature classes from our ablation study, proximity and discourse / topicality, have very different average feature weights. On the one hand, proximity is highly reliable, with both distance and depth features achieving weights over 0.1 despite modelling similar phenomena. We note that depth has both the higher average feature weight and smallest number of distinct features, suggesting, in the least, that position in the self-ordering forest is meaningful for informing resolution. On the other hand, discourse / topicality has a large number of low-weighted features. While it could be that a large number of dimensions required to model these phenomena, it would be instructive to investigate ways to compress the space, e.g. by binning. We note that within this class, lifespan score performs well, having 441 features with average weight 0.100, suggesting that modelling singletonhood is an important sub-task of coreference resolution.

Despite its strength in our ablation study, lexicalised features have low weights as a class. This is explained by feature set sparsity: text based features comprise 90% of lexicalised features but have lower weight on average than the more compact POS tag and word shape features. While removing text-based lexicalised features from the LIMERIC model results in a small performance decrease, they could be profitably omitted from systems for which short computation time is vital.

### **Profile of the Forest**

The above analysis demonstrated the importance of proximity features and suggested that the cognitively-motivated depth was perhaps a better indicator than textual distance. Given the importance we give to the relative depth of an entity later in this thesis

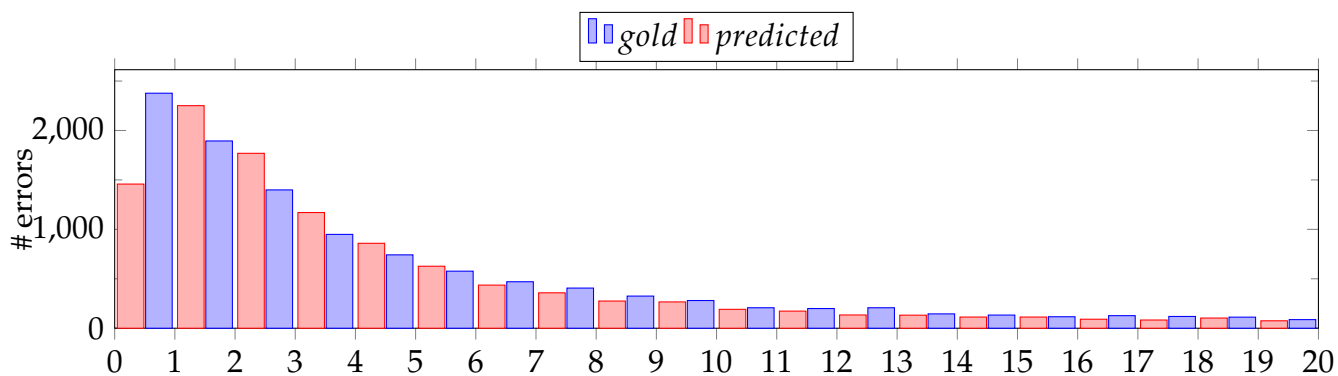


Figure 4.5: Depth in forest of correct prediction in CoNLL-2012 DEV using gold preprocessing.

(cf. Chapter 6), we explore its profile in Figure 4.5, which plots the depth from the right frontier of the forest of the correct target of a reduce operation in CoNLL-2012 DEV (gold preprocessing). The blue series represents the depth of this target when clustering follows gold standard transitions and the red series depths when following system predictions from the full LIMERIC model above. We note a very long tail to this distribution and plot up to depth 20, which cumulatively represents 88% and 89% of the gold and system transitions data, respectively.

We can see that, consistent with our design to keep accessible targets at the right frontier of the forest (depth 0), the majority of the correct targets are at small depth values and the distribution decays quickly away from this point. However, the peak in the distribution is at depth 0 for gold transitions but depth 1 for predicted transitions and predicted targets tend to be deeper in the forest than gold targets.

This difference between the distribution in the two settings is discouraging given that we tune feature weights based on gold transitions in training, while runtime follows system predictions. While training on predicted transitions using beam search would address this problem, Björkelund and Kuhn (2014) dismiss this approach since the coreference feature set is not sufficiently informative to prevent the correct resolution of a document quickly falling out of the beam, requiring it to be repeatedly re-seeded

from the gold standard. We use these observations to motivate including both distance and depth features in our feature set despite their similarity: if depth is imperfectly tuned, its negative impact can be countered by distance.

## 4.4 Evaluation

We benchmark the performance of LIMERIC against the current state of the art for the CoNLL-2012 shared task guidelines (cf. Chapter 2). All experiments are run using the standard splits of the OntoNotes 5 dataset, version 8.01 of the official scorer<sup>3</sup> (Pradhan et al., 2014), and evaluate performance using the CoNLL metric which averages the MUC F-score (Vilain et al., 1995), B<sup>3</sup> (Bagga and Baldwin, 1998) and CEAFE (Luo, 2005). Gold and automatic pre-processing corresponds to the annotations provided for these settings in the official dataset release (cf. Chapter 2). Error analysis follows using Kummerfeld and Klein’s (2013) software release.

### 4.4.1 Benchmarking

We compare our performance against that of three systems which reflect the diversity of state-of-the-art approaches introduced in Chapter 3. The performance of Lee et al.’s (2011) multi-pass sieve architecture has been surpassed by more recent systems, but is included as a reference entity-level approach. Fernandes et al. (2012), Björkelund and Kuhn (2014), and Chang et al. (2013) all use structured prediction whose mention-synchronous decoding incorporates global consistency constraints in a similar way to best-first decoding in our mention-pair models. Fernandes et al. uses no entity-level features and Björkelund and Kuhn’s software release can be used with or without such features. We compare against both settings.

Table 4.5 presents our performance on the CoNLL-2012 TEST dataset, with gold and automatic preprocessing. Models are trained on the concatenation of the TRAIN and

---

<sup>3</sup><http://conll.github.io/reference-coreference-scorers/>

System			Gold				Auto			
	G	E	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
Lee et al. (2011)		✓	-	-	-	-	63.82	51.21	47.60	54.21
Fernandes et al. (2012)	✓		72.18	59.17	55.72	62.36	70.51	54.47	53.86	60.65
Björkelund and Kuhn (2014)	✓		72.65	59.98	57.94	63.52	69.25	56.14	54.19	59.86
Chang et al. (2013)	✓	✓	-	-	-	-	69.48	57.44	53.07	60.00
Björkelund and Kuhn (2014)	✓	✓	73.80	62.00	59.06	64.95	70.72	58.58	55.61	61.63
LIMERIC	✓	✓	73.83	60.70	58.13	64.22	70.09	56.21	53.68	59.99

Table 4.5: Performance of LIMERIC on CoNLL-2012 TEST.

DEV portions of the dataset. G denotes where systems use non-local decoding and E where systems use entity-level features.

We can see that, despite the simplicity of our learning and decoding compared to structured predication, our system compares favourably with existing systems. In the gold preprocessing setting, we outperform by at least 0.70% CoNLL score all systems which use only global-consistency decoding or entity-level modelling, arguing for their mutual benefit to the task. Furthermore, we are competitive with the contemporary Björkelund and Kuhn (2014) system, which at the time of this work, was the best reported performance on CoNLL-2012<sup>4</sup>.

The transition from gold to automatic preprocessing is more problematic for LIMERIC than the other systems. While we compare favourably with Lee et al. and Björkelund and Kuhn (without entity-level features), and Chang et al., we trail Björkelund and Kuhn’s best system by 1.64%. We saw in the last section that our system assigns high weights to grammatical, attribute cohesion, and POS tag features, all of which will be noisy in automatically pre-processed data. We explore features to improve our performance across both settings in following chapters.

<sup>4</sup>As noted in Chapter 3, the current best reported performance is Wiseman et al. (2015)

### 4.4.2 Error Analysis

We explore system performance further in Figures 4.6 and 4.7. These plots show the number of errors made by LIMERIC, as well as both configurations of Björkelund and Kuhn’s (2014) IMS system, in the seven error categories reported by the tool described by Kummerfeld and Klein (2013). Local denotes where IMS does not use entity-level modelling and global where it does.

Our first observation is that the error profiles of LIMERIC and Björkelund and Kuhn look similar: both systems make a large number of conflated entity and divided entity errors, comparatively few missed mention and missed entity errors, and fewest span, extra mention, and extra entity errors.

Despite IMS (global) outperforming LIMERIC, we generate fewer errors in five out of seven error categories: span, conflated entity, missed mention, extra mention, and extra entity. However, the two error categories where IMS is stronger, divided entity and missed entity, are shown in Kummerfeld and Klein to have the biggest impact on standard evaluation metrics. These biases are reasonable given that the metrics have been designed to measure how good produced clusters are and being overly conservative means that document cohesion has not been understood.

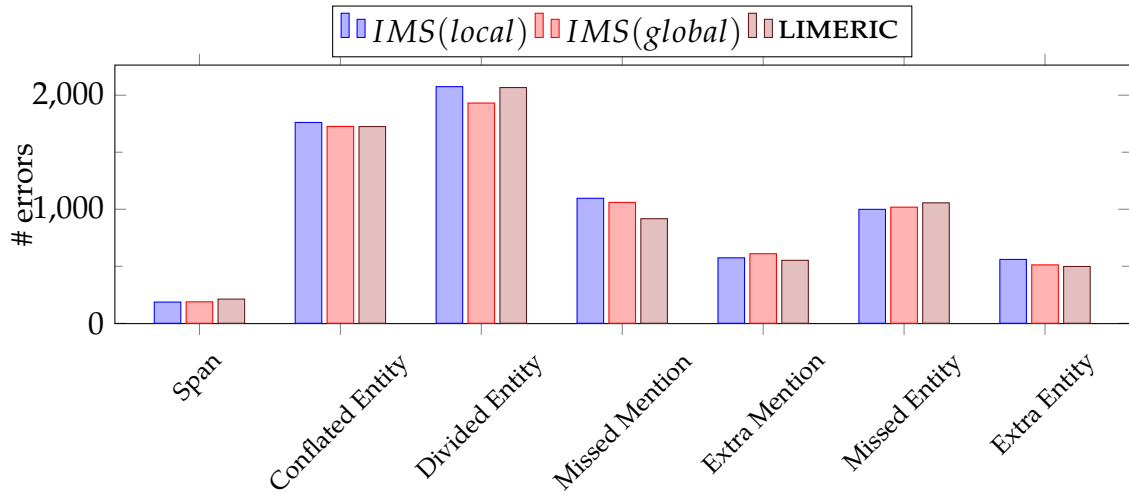


Figure 4.6: Errors made by LIMERIC and the current state of the art, IMS, on CoNLL-2012 TEST using gold preprocessing.

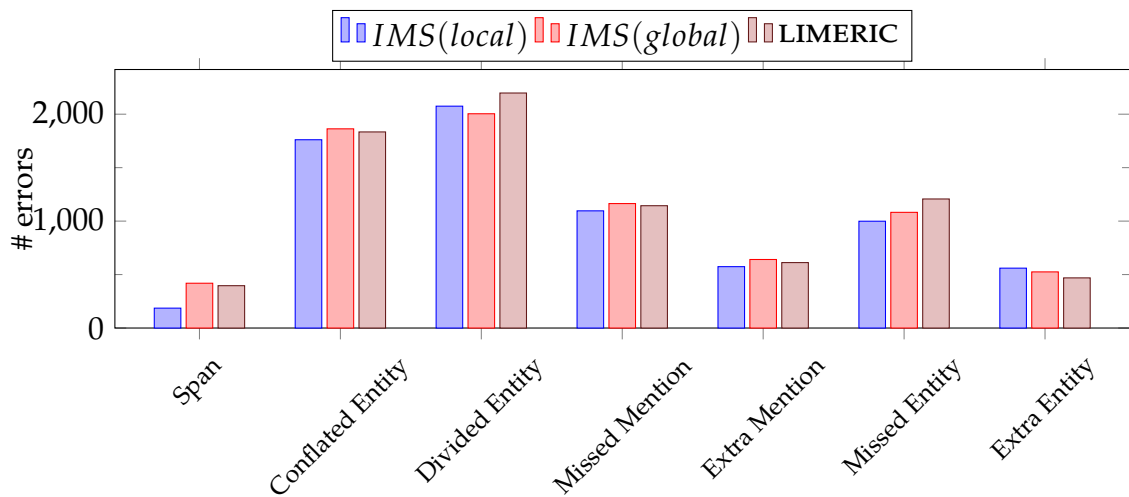


Figure 4.7: Errors made by LIMERIC and the current state of the art, IMS, on CoNLL-2012 TEST using automatic preprocessing.

## 4.5 Summary

In this chapter, we have designed and implemented **LIMERIC**, an incremental coreference resolution engine based on insights from the *LR*(1) shift-reduce parsing algorithm and cognitive models of human discourse processing. **LIMERIC** processes a document by reading extracted mentions in top-to-bottom, left-to-right human reading order, without look ahead. Entity clusters emerge as a discourse progresses, with growing clusters being stored in a self-ordering list which operates as a simple model of the human mind. As well as being linguistically motivated, our formulation gives us a natural way to encode both non-local decoding and entity-level modelling, and outperforms all documented systems using just one encoding of global consistency in isolation.

We implement a rich feature set based on our review of the literature. We formulate general processes to convert existing mention-pair indicators into entity-level features, and propose lifespan score and depth as novel, cognitively-aware ways to model singleton detection and perceived proximity. Analysis validates the soundness of these proposals and reveals that discourse patterns are particularly well-learned. We therefore extend our discourse model in the next chapter, using insights from the Accessibility hierarchy. Our analysis also illustrates that system performance gains from cohesion features plateau as they broaden to capture fuzzier relationships. This observation is studied in Chapter 6 and frame semantic inference features to address it are explored in Chapter 7.

We benchmark our system against the contemporary state of the art and find that despite its simplicity, it is competitive with such. Error analysis shows that it trails state of the art from being more conservative.

## 5 Accessibility Hierarchy

*Work described in this chapter forms part of the conference paper Kellie Webster and Joel Nothman. 2016. Using mention accessibility to improve coreference resolution. In Proceedings of the 54th Annual Conference of the Association of Computational Linguistics.*

In this chapter, we build a richer discourse model into LIMERIC to capitalise on our finding that our competitive performance is strongly associated with our ability to learn discourse transition patterns.

We start by formalising our experimental questions before adapting the Accessibility hierarchy to the written English data of OntoNotes 5.0. We then confirm the relevance of the fine-grained typing scheme to annotations in the dataset in two ways. First, we analyse discourse trends across OntoNotes through the lens of the hierarchy. Second, we devise a series of experiments that extend the discourse modelling of LIMERIC to include the new fine-grained mention types.

Feature prefixes using type transitions boost the performance of LIMERIC to be as strong or stronger than the state of the art set by Fernandes et al. (2012) on the shared task, achieving 64.96% and 60.58% on CoNLL-2012 using gold and automatic preprocessing, respectively. We attribute our significant improvement to our modelling of fine-grained trends in reference expression usage which cannot be formulated in the commonly used, coarse-grained typology of mentions.

*Full name + modifier < Full name < Long definite description < Short definite description*  
*< Last name < First name < Distal demonstrative + modifier < Proximate demonstrative + modifier*  
*< Distal demonstrative + NP < Proximate demonstrative + NP < Distal demonstrative*  
*< Proximate demonstrative < Stressed pronoun + gesture < Stressed pronoun < Unstressed pronoun*  
*< Cliticised pronoun < Verbal inflections < Zero*

Figure 5.1: Accessibility hierarchy of Ariel (2001).

## 5.1 Research Questions

In Chapter 3, we saw that the Accessibility hierarchy is one of two mechanisms used by Ariel (2001) to explain human reference resolution. The hierarchy orders a series of fine-grained classes of reference expression according to the level of activation their discourse referent is expected to be. It is these mention types, via their position in the hierarchy, which instruct hearers how to retrieve referent discourse entities, thus guiding our resolution of coreference. The hierarchy given in Ariel, derived for spoken Hebrew, is reproduced in Figure 5.1.

In this chapter, we analyse OntoNotes coreference annotations and design a prefixing strategy to incorporate insights from the hierarchy into LIMERIC’s discourse model. Both of these contributions address two research questions about the applicability of the Accessibility hierarchy to coreference resolution. Our research questions are:

- 1) Do the fine-grained classes of the Accessibility hierarchy provide a better description of English coreference than currently used coarse-grained classes?
- 2) Does the total ordering given in the Accessibility hierarchy describe coreference patterns in English?

In this chapter, we find strong evidence for the applicability of the fine-grained classification scheme, but only weak evidence for its proposed ordering. We suggest that there is not strong evidence for the proposed ordering of reference *forms* since

different forms have different *functions* in texts and it may be that ordering pertains to functions only.

## 5.2 Experimental Setup

In this section, we describe our implementation of the Accessibility hierarchy and how we use it to describe discourse transition pairs, which are the base unit of our analysis of discourse trends in OntoNotes, and are used as prefixes in the feature development experiments which follow.

### 5.2.1 Mention Classification

Our experiments start by classifying mentions as belonging to a particular class from the Accessibility hierarchy. To do this, we first map the hierarchy to the simple ordinal numbering scheme given in Table 5.1. In defining this mapping, we necessarily make some changes to Ariel’s classes so that we can describe English text, but aim to keep these minimal. Table 5.2 illustrates our classes by giving the most common mention strings for each.

We have generalised last name and first name to single-word name ( $AR = 7$ ) and full name to multi-word name ( $AR = 2$ ) to remove the implicit assumption that we are only dealing with person entities. This mapping, however, does not account for entities whose full name is one token long, such as some organisations and geopolitical entities. Indeed, the most common mentions strings in  $AR = 7$  are single-token entity names. However, the full task of identifying canonical names would require named entity linking, which is outside the scope of this work.

Ariel does not make a distinction between first, last, or full names with modifier, so we similarly allow the  $AR = 1$  class to incorporate single- or multi-word names accompanied by modification. Modifiers to name mentions are any tokens not tagged with the same NER label as the head token, disregarding determiners, possessive

AR	Description	%
1	Name + modifier	5.1
2	Multi-word name	8.9
3	Long indefinite description	22.2
4	Short indefinite description	9.4
5	Long definite description	11.3
6	Short definite description	7.5
7	Single-word name	11.2
8	Distal demonstrative + modifier	0.2
9	Proximate demonstrative + modifier	0.01
10	Distal demonstrative + NP	0.7
11	Proximate demonstrative + NP	1.2
12	Distal demonstrative	0.8
13	Proximate demonstrative	0.6
14	Pronoun	21.0
-	Zero	-

Table 5.1: Accessibility rank values used in our experiments, with their base distribution over extracted NPs.

AR	
1	Mr. Keating; President Bush; President Clinton; Mr. Clinton; Mr. Papandreou
2	Hong Kong; New York; the United States; last year; Xinhua News Agency
3	real - estate; national service; program trading; many people; foreign capital
4	there; people; all; anything; everything
5	the SAR government; the same time; the Bush administration; our country 's mainland; the Korean peninsula
6	the world; the people; the president; the company; the market
7	Taiwan; first; Jesus; God; today
8	all that; those responsible; those who 'S'
9	this : 'S' ; this : 'S'; this to say; this , 'S'
10	that time; those people; those days; that guy; that way
11	this case; this point; these things; this guy; these people
12	that; those
13	this; these
14	it; I; you; they; he

Table 5.2: Most common mention strings for each accessibility rank value.

markers, and punctuation. Table 5.2 shows examples of person names with role and title information, though modification can also include more complex structures such as apposition. We therefore expect the distinction between  $AR = 1$  and other proper name classes to be clouded when using automatic preprocessing.

We have introduced classes for indefinite descriptions since definiteness is an important grammatical distinction for English, though not for Hebrew. We opt to insert indefinite descriptions above definite descriptions since indefinite descriptions are more likely to introduce discourse entities than definite descriptions are in OntoNotes (see Table 5.3). We label any mention started by the determiner ‘*the*’ or a possessive pronoun as a definite description, and any nominal not started by one of these articles or a demonstrative, including those started by the determiner ‘*a*’ or no determiner at all, as an indefinite description.

We label descriptions as long or short by according to the number of tokens they comprise when possessive markers, punctuation, and articles are excluded. Short descriptions are those where only one token, the mention’s head, remains while long descriptions are anything longer than this. In Table 5.2, we see the extra tokens can cover noun compounding, adjectival pre-modification, and possessive constructions. Outside these common examples, they also cover prepositional phrase post-modification.

Distal demonstratives are mentions starting with ‘*those*’ or ‘*that*’ and proximate demonstratives are those starting with ‘*these*’ or ‘*this*’. Modification to a bare demonstrative is called an NP ( $AR = 10$  and  $11$ ) if the POS tag of its syntactic head starts with ‘*N*’ and a modifier ( $AR = 8$  and  $9$ ) otherwise. The most common modifiers are clauses.

Also given in Table 5.1 is the base distribution over extracted mentions. Over one-third of extracted mentions are indefinite descriptions, while proper names and pronouns each make up roughly one-quarter of mentions. The remainder is mostly definite descriptions, though the other mention types have scattered representation.

### 5.2.2 Discourse Transition Pairs

Next, we would like to know what classes of mention tend to co-occur in coreference relationships. In our analysis of OntoNotes, we do this by iterating over entity clusters and tracking the classes of the mentions these cover. To form prefixes which reflect co-occurrence tendencies, we consider the classes which would similarly be related if the current mention were to join a given candidate antecedent cluster.

We define discourse transitions to be tuples of  $AR$  values over coreferential mentions. This means that trends will surface as commonly seen tuples. In the following excerpt, we could generate the 3-tuple (1, 14, 14) for the discourse transition across the cluster of the three coreferential mentions indicated in bold. Defining such arbitrarily large tuples is problematic given that sparsity would increase with tuple length and consistencies in regions of large clusters might not necessarily emerge. It also limits the applicability of transitions to incrementally growing clusters. We therefore reduce tuples to be pairs since mention-pair models have been important in coreference research, and the entity-level modelling we use in this thesis is based implicitly on mention-pair features.

**Israeli Prime Minister Ehud Barak** <sub>$AR=1$</sub>  called **his** <sub>$AR=14$</sub>  cabinet into special session late Wednesday , to discuss what **he** <sub>$AR=14$</sub>  called a grave escalation of the level of violence in the Palestinian territory.

## 5.3 Trends in OntoNotes 5.0

Using our implementation of mention classification and discourse transition pairs, we are now ready to explore trends in OntoNotes, specifically the CoNLL-2012 DEV portion of the dataset. The goal of this analysis is to identify any consistent trends in the discourse behaviour of extracted mentions according to their assigned fine-grained class. That is, we would like to find if there are any rules of thumb akin to “full names introduce entities, pronouns are anaphoric” that we can formulate over our

fine-grained types. We find that the discourse behaviour of proper names and nominals shows systematic trends in our fine-grained typology which are not expressible in a coarser-grained typology. While we find a tendency for referential forms to increase in accessibility across clusters via pronominalisation, we also find a tendency for accessibility to be retained. More problematically, we find no clear tendency for definite descriptions to condition a reduction, retention, or even augmentation of accessibility: their discourse behaviour is more complex than given in the Accessibility hierarchy.

We also consider tendencies in the  $AR$  values of extracted mentions which cannot be aligned to gold mentions, since these correspond to discourse singletons, which we would also like to characterise. We similarly find that a fine-grained classification scheme is better than a coarse-grained one to describe tendencies in discourse singletonhood.

### 5.3.1 Discourse Transition Trends

To extract discourse transition pairs over the coreference annotations in DEV, we iterate over the entity clusters; for each mention in each cluster, we generate up to three pairs, one for each of its closest antecedents. For instance, for the third mention ‘*he*’ in our example above, we generate the two pairs (1, 14) and (14, 14).

Extracting multiple pairs for each mention enables us to capture the insight described of entity-level models that anaphoric links may be latent at the entity level. Table 5.3 aggregates the relative frequency of these tuples, with  $AR(antecedent)$  on the vertical and  $AR(anaphor)$  on the horizontal, with values less than 5% omitted for clarity. The first column gives the proportion of cluster-initial mentions of each  $AR$  type (e.g. 21% of gold clusters have a long definite description as their first mention). Subsequent proportions in each row are normalised to sum to 1. No values are given for  $AR = 9$  due to its low count (9 instances).

Given the normalisation applied to the rows of Table 5.3, each row indicates the probability distribution for the expected next mention of a cluster. In the representation

$AR(anaphor) \backslash AR(antecedent)$			1	2	3	4	5	6	7	8	9	10	11	12	13	14
Name + modifier	1	0.12	0.22	0.06					0.15							0.48
Multi-word name	2	0.12		0.31				0.06	0.14							0.40
Long indefinite description	3	0.21			0.07		0.09	0.14								0.52
Short indefinite description	4	0.06				0.05		0.12	0.05							0.65
Long definite description	5	0.14					0.21	0.15	0.09							0.41
Short definite description	6	0.08					0.07	0.37	0.07							0.39
Single-word name	7	0.15							0.49							0.42
Distal demonstrative + modifier	8	0.01			0.05			0.05		0.05						0.79
Proximate demonstrative + modifier	9	0.01														
Distal demonstrative + NP	10	0.01					0.07	0.10				0.13				0.54
Proximate demonstrative + NP	11	0.02					0.05	0.10	0.11				0.12			0.54
Distal demonstrative	12	0.00						0.05	0.05					0.34		0.43
Proximate demonstrative	13	0.00							0.08				0.05		0.05	0.71
Pronoun	14	0.08							0.09							0.82

Table 5.3: Accessibility transitions in CoNLL-2012 DEV by accessibility rank value.

of Table 5.3, the rule of thumb “full names introduce entities, pronouns are anaphoric” translates to an expectation that the rows of proper names ( $AR = 1, 2$ , and  $7$ ) will have high probability mass in higher accessibility forms, while pronouns ( $AR = 14$ ) should have high probability mass down all rows.

Reading the rows for proper name types, we can see that modified and multi-word names have a tendency to reduce to single-word names, and both reduce to pronouns. Single word names retain their mention form and reduce to pronouns with roughly equal probability. Both these observations are consistent with our expectations.

Pronouns similarly behave as expected. There is a band of dark shading in the pronoun column indicating that all mention types reduce to be pronouns. That is, pronouns can have the function of indicating coherence by making anaphoric reference to any mention type. Furthermore, once reference has reduced to be pronominal, there is a high likelihood (82%) that this form will be retained. We note also that this trend

reflects the predictions of Centering theory, which says that topical entities tend to be referred to with chains of pronouns, as their salience is retained throughout a discourse.

Since mentioning a discourse entity will increase its accessibility in the mind of a reader, we might expect *AR*s to increase between antecedent and anaphor. If this were the case, we should see more shading above the diagonal than below. Aggregating over OntoNotes transition pairs, 22% of transitions increase *AR* while 14% of transitions decrease *AR*. That is, while there is a preference for moving up in rank rather than down, this preference is slight. 64% of transitions involve accessibility being retained.

Stronger evidence against a tendency for accessibility to increase over references is the band of shading we see down the rows for *AR* = 5, 6, and 7. This shading means that definite descriptions can validly refer to mentions of any type, as we saw for pronominal reference. This is reasonable since in OntoNotes documents, particularly those from the news domain, mentions like ‘*the company*’, ‘*the nation*’, ‘*the city*’, and ‘*the X-year-old*’ appear to have the same discourse function as pronouns, acting as conventionalised quasi-pronouns, injecting extra facts about important entities in short spans. This banding also adds to the list of reasons why definite descriptions have been so problematic for modern resolution systems: as well as being semantically rich, their discourse behaviour is complex.

Finally, we assess whether the transitions we see in Table 5.3 are expressible in the traditional coarse-grained typology of coreference mentions. Our fine-grained typology differs from this standard in at least two dimensions: mention classes reflect the length of mentions, and nominal mentions are further classified by their article. We see that both these dimensions are important for understanding the discourse behaviour described in Table 5.3.

First, article is important. Long indefinite descriptions are more likely to start coreference clusters than long definite descriptions (21% vs. 14%), which are in turn much more likely to start clusters than demonstratives. Also, length is important because short indefinite descriptions are more likely to reduce to pronouns than long

definite descriptions. Also, short definite descriptions have a higher chance of being retained throughout the discourse than long definite descriptions. When it comes to whether the actual mention string is retained short definite descriptions have a higher tendency for surface form to match than long definite descriptions do: 86% of short definite descriptions are head matched, compared to 60% of long definite descriptions, and 60% of short definite descriptions are string matched, compared to 27% of long definite descriptions.

### 5.3.2 Anaphoricity Trends

We now consider extracted mentions which are not aligned to gold mentions. We have seen previously (cf. Chapter 4) that these correspond to discourse singletons, first mentions of an entity which are not mentioned again in a discourse. Given that we learn coreference jointly with singleton identification, we would like to understand any patterns in singletonhood by *AR* value, since these can potentially improve our ability to label mentions as markable or not. Table 5.4 gives the proportion of unaligned extracted mentions by *AR* value.

We can see that most mention types have a high proportion of singletons, presumably due to our high recall implementation of mention extraction. Pronouns are the mention type with the lowest likelihood of being singletons, which accords with our expectation that their function is largely anaphoric. Where pronouns are in singleton clusters, they tend to be second person (*'you'*) and third person neuter (*'it'*) or, less commonly, first person plural (*'we'*) pronouns. This makes sense given that these pronouns have a rhetoric, non-referential function which falls outside the scope of OntoNotes annotation. We also remember that texts from new genres (e.g. telephone conversation) were artificially sectioned in creating the OntoNotes corpus. At times, this sectioning means that anaphoric pronouns are not annotated as such since their antecedent is not in the same section as the pronoun.

AR	Description	
1	Name + modifier	0.56
2	Multi-word name	0.65
3	Long indefinite description	0.89
4	Short indefinite description	0.92
5	Long definite description	0.75
6	Short definite description	0.54
7	Single-word name	0.44
8	Distal demonstrative + modifier	0.69
9	Proximate demonstrative + modifier	1.00
10	Distal demonstrative + NP	0.43
11	Proximate demonstrative + NP	0.41
12	Distal demonstrative	0.43
13	Proximate demonstrative	0.60
14	Pronoun	0.21

Table 5.4: Proportion of singletons in CoNLL-2012 DEV by accessibility rank value.

Along with demonstratives, proper names are the type with the next lowest proportion of singletons. Single word names are less likely to be singletons than modified and multi-word names. This may be due to at least two different factors. The first is the presence of non-markable names among our set of singletons. In particular, proper names in an appositional phrase are not markable. The second is that the burden of supplying disambiguating modification will be more worthwhile for entities which are important in the discourse and mentioned multiple times. That is, our statistics reflect common sense intuitions about language use, but are not expressible in a coarse grained mention typology typically used in approaches to coreference resolution.

Both indefinite description types, as well as being the most common type of extracted mentions, show the highest proportions of discourse singletons. Exploring this case further, we again find fine-grained patterns in mention length and article, though they mirror the pronominalisation pattern described previous. We now see that the likelihood for an indefinite description to form a singleton cluster is independent of its length and is uniformly high. On the other hand, long definite descriptions are more likely than short definite descriptions to form singleton clusters. This is consistent with

the finding in Recasens et al. that indefinite NPs are more likely than quantified NPs to form discourse singletons, and that the chance of an NP forming a discourse singleton increases with the number of modifiers. Since length and article are the key sub-typing factors in the Accessibility hierarchy, this is good evidence in favour of the hierarchy’s fine-grained classification. That is, as well as helping us to understand the shape of entity clusters, the fine-grained mention types in the Accessibility hierarchy are helpful for understanding patterns in the anaphoricity of mentions.

In terms of our second research question on type ordering, we do not necessarily anticipate any linear patterns according to the *AR* values of mention types.

## 5.4 Evaluation

In this section, we formulate novel features with reference to the preceding analysis and test their usefulness by introducing them into LIMERIC. In so doing, we find further evidence in favour of the Accessibility hierarchy’s fine-grained typology, but only marginally in favour of its ordering.

### 5.4.1 Feature Prefixes

In our discourse modelling for LIMERIC, which we found was an important factor contributing to our competitive performance, we used three variants of each extracted feature: one unprefixed, one prefixed with the current mention’s coarse-grained type (name, nominal, or pronoun), and one prefixed with the concatenation of the types

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
LIMERIC	73.24	61.29	59.07	64.53	69.14	56.59	54.91	60.21
<i>AR</i> Transitions	73.80	61.98	60.26	65.35	69.60	57.06	55.53	60.73
<i>AR</i> Rankings	73.32	61.34	59.36	64.67	69.08	56.69	55.00	60.26

Table 5.5: Performance of *AR* feature prefixing on CoNLL-2012 DEV.

of the current and closest antecedent mention in the proposed entity cluster. In this section, we test our research questions directly by introducing yet a fourth variant with a prefix based on the richer model of the Accessibility hierarchy.

In particular, we experiment with two implementations of this prefix, namely *AR* Transitions and *AR* Rankings. *AR* Transitions leverages the fine-grained classification scheme of the hierarchy, while *AR* Rankings assumes its priority structure. We find that *AR* Transitions outperforms *AR* Rankings in both gold and automatic preprocessing settings. Furthermore, features learned in our *AR* Transitions model illuminate interesting discourse patterns not modelled in our baseline LIMERIC system. We therefore interpret our results in favour of our first research question, though assuming the Accessibility hierarchy’s ordering does not diminish performance.

### ***AR* Transitions**

In *AR* Transitions, this third prefix is a fine-grained version of our second, transition-pair prefix. It is formed by concatenating the *AR* value of the current mention with that of the closest antecedent in the candidate entity cluster. This means that our discourse prefixes now represent three levels of generalisation: features can indicate whether they describe coreference behaviour across of all comparisons (unprefixed), across coarse-grained types (first and second prefixes), and, now, across fine-grained types (our new third prefix). Since this prefix expands our possible feature set by a factor of  $14^2 = 196$ , we opt not to also introduce a fine-grained prefix analogous to the first coarse-grained prefix, and instead allow LIMERIC to learn such patterns sparsely via our new transition-pair prefix.

We can see in Table 5.5 that, despite the potential for making our feature set overly sparse, we gain 0.82% and 0.52% using gold and automatic preprocessing. This improvement is from a simultaneous boost in precision and recall, with precision increasing 0.94% and 0.49% and recall increasing 0.71% and 0.53%. That is, *AR* Transitions make

Class	LIMERIC		AR Transitions		Change	
	Features	Weight	Features	Weight	Features	Weight (%)
Grammar	1746	0.119	3913	0.063	+2.2	-53
Surface Cohesion	2713	0.131	8381	0.067	+3.1	-51
Attribute Cohesion	8697	0.134	19982	0.063	+2.3	-47
Lexical Cohesion	3326	0.068	6935	0.033	+2.1	-49
Proximity	3363	0.113	8824	0.075	+2.6	-66
Discourse / Topicality	4964	0.047	9252	0.032	+1.9	-68
Lexicalised	371489	0.067	742385	0.034	+2.0	-51

Table 5.6: Number of distinct features and their average weight in our *AR Transitions* model, compared to LIMERIC.

the resolver more discriminative to make correct decisions, as well as promoting new matches.

To understand the differences between our LIMERIC and *AR Transitions* models, we calculate the average weights by feature class, as we did in the previous chapter. We reproduce the feature weights from LIMERIC in the leftmost section of Table 5.6 and compare those with those of our *AR Transitions* model from gold preprocessing. The rightmost section then tabulates the change in number of features and average weights, determined to be the percentage the new *AR Transition* weight is of the previous LIMERIC weight.

Firstly, we can see that the number of distinct features has only inflated by a factor of between 1.9 and 3.1, rather than the potential factor of 196, indicating that fine-grained modelling is only important for certain transitions. We can see that most feature families have dropped in average weight by about half, as weight is spread across the extra level of granularity we now have. We note two interesting cases. The number of surface cohesion features increases threefold, yet the average weight of each is only roughly half what it was in the LIMERIC model: sub-typing by *AR* allows surface cohesion to be a more important feature class overall. On the other hand, the number of proximity features only reduces by a factor of 1.9 but the change in average magnitude is 68%:

sub-typing is less important for learning patterns with mention distance, and the model changes minimally to reflect this.

We further explore these changes in Tables 5.7 and 5.8 show the ten most positively and most negatively weighted of our new features. We can see that the most represented mention type among the highly positive features is single word names ( $AR = 7$ ). This makes sense given that they were a mention type with a low likelihood of being discourse singletons and, as an anaphor, were valid anaphors to almost all mention types. Pronouns were another mention type with a high likelihood of being anaphoric and were valid anaphors for all mention types. However, there are only two highly weighted features for pronouns in Table 5.7. This is perhaps because they are mentions for which proximity is a key indicator of reference resolution, which has diminished performance in this model.

Interestingly, in the important pronoun features, the pronoun ( $AR = 14$ ) is in the antecedent position, which is not its canonical role according to our coarse-grained rule of thumb. The example feature for the transition from pronoun to single-word name says the link is likely when the pronoun is clustered with a name matching the current mention. Such a feature crucially relies on our entity-level modelling of the task, and its importance shows that accessibility should not be assumed to uniformly decrease through a cluster.

While the features in Table 5.7 were commonly attested among coreferential mentions, all features in Table 5.8 are generally rare, and the examples in this table give us some indication why this is the case. Among proper name mentions, multi-word names ( $AR=2$ ) which agree on only one word are unlikely to be coreferential. This feature is particularly insightful for person names since it separates family members such as Bill and Hillary who share a surname, Clinton, when mentioned as '*Bill Clinton*' and '*Hillary Clinton*'. Coreferential instances which are ruled out by this heuristic include entities with alternative names, or alternative spellings, of names. This valuable feature crucially relies on the fine-grained classification of the Accessibility hierarchy, and is

Transition <i>antec ana</i>		Feature	Weight	Example
14	7	1 common word	1.17	'President Kostunica ... <u>He</u> ... <u>Kostunica</u> '
5	7	1 common word	0.92	'the dominant republic of Serbia ... <u>Serbia</u> '
2	7	1 common word	0.88	'The Chicago Tribune ... <u>Tribune</u> '
1	7	next sentence	0.84	'Slobodan Milosevic met with Russian Foreign' 'Minister <u>Igor Ivanov</u> . <u>Ivanov</u> says ...'
14	1	same sentence	0.82	'"Frankly, I missed my family," said <u>Mr. Rosenblatt</u> .'
7	7	same sentence	0.79	'Then give to <u>Caesar</u> what belongs to <u>Caesar</u> '
2	7	acronyms	0.75	'the Socialist Party of Serbia ... <u>SPS</u> '
7	5	length <sub>ana</sub> =3	0.69	'the Clintons ... the first couple'
3	1	first token <sub>antec</sub> 'CD'	0.68	'one Merc broker ... Mr Dubnow'
1	7	1 common word	0.68	'President Kostunica ... <u>Kostunica</u> '

Table 5.7: Ten most positively weighted features in our AR Transitions model.

not expressible when all proper names are considered equivalent. Indeed, single-word and modified names sharing one token was a highly positive indicator for coreference and collapsing all names would neutralise this polarity we see.

Within a sentence, there are multiple discourse transitions which are dispreferred: Long definite descriptions ( $AR=5$ ) are unlikely to be coreferential with other close long indefinite descriptions, long definite descriptions ( $AR=3$ ), or single-word names. The former is presumably because such repetition is cumbersome, and one mention being long should be sufficient to give readers any necessary information. Cases where this dispreferred construction is licensed includes poly-clausal sentences, and when one mention is embedded as a modifier in another reference expression. Again, single word names in the same sentence was a highly positive indicator of coreference, where single-word names and modified names behave polar to one another.

Overall, the discourse patterns described by our new features are complex and their explanation requires the Accessibility hierarchy's fine-grained classes. That is, by including feature prefixes based on a mention typology, we are able to learn a richer

Transition <i>antec ana</i>		Feature	Weight	Example
14	7	0 common words	-1.01	'the Phillies ... <u>their</u> '
5	7	0 common words	-0.97	'the Israeli state ... Israel'
3	5	same sentence	-0.97	'When you have a <u>malignant tumor</u> , you' 'may remove <u>the tumor itself</u> surgically.'
1	7	same sentence	-0.90	'He promises to bring <u>Mr. Milosevic</u> to justice and' 'rid the police and judiciary of <u>Milosevic</u> loyalists.'
5	5	same sentence	-0.87	'it would pay attention to <u>the situation on the Korean</u> ' ' <u>peninsula</u> and sincerely hoped that <u>the situation</u> ' ' <u>on the Korean peninsula</u> would be relaxed ...'
2	2	1 common word	-0.82	'Fitty Cent ... <u>Fifty Cent</u> '
2	7	0 common words	-0.80	'National Ice Hockey League ... NHL'
3	3	length <sub>ana</sub> =3	-0.78	'Miami Dade and Palm Beach counties ... both two counties'
7	7	prev word <sub>antec</sub> =in	-0.76	'There are 26 insurance companies now in <u>China</u> ' 'and more than one hundred overseas insurance companies' 'that have established administrative organizations in <u>China</u> .'
5	7	same sentence	-0.76	'The sales drop for <u>the No. 1 car maker</u> may have been' 'caused in part by the end in September of dealer incentives' 'that <u>GM</u> offered ...'

Table 5.8: Ten most negatively weighted features in our AR Transitions model.

Class	LIMERIC		AR Rankings		Change	
	Features	Weight	Features	Weight	Features	Weight (%)
Grammar	1746	0.119	1900	0.103	+1.1	-87
Surface Cohesion	2713	0.131	2920	0.118	+1.1	-90
Attribute Cohesion	8697	0.134	9541	0.115	+1.1	-86
Lexical Cohesion	3326	0.068	3959	0.056	+1.2	-82
Proximity	3363	0.113	3496	0.100	+1.0	-88
Discourse / Topicality	4964	0.047	5376	0.040	+1.1	-85
Lexicalised	371489	0.067	425335	0.055	+1.1	-82

Table 5.9: Number of distinct features and their average weight in our *AR* Rankings model, compared to LIMERIC.

model for coreference than is possible with just a coarse-grained mention typology, or in a simple rule that accessibility should increase in certain environments.

### AR Rankings

In *AR* Rankings, the third prefix takes one of three values on an anaphor-candidate antecedent cluster pairing, with the choice reflecting whether the *AR* of the current mention is greater than, equal to, or less than that of the closest mention in the cluster. These features allow us to collapse the sparsity of *AR* Transition prefixes, but rely on the Accessibility hierarchy being viewed as a priority structure, rather than a fine-grained classification scheme of mention types. The *AR* Rankings model performs similarly to LIMERIC, boosting CoNLL by just 0.14% and 0.05% on the gold and automatic preprocessing settings. The disappointing performance is consistent with our resource analysis, which showed that *AR* values do not uniformly decrease throughout an entity cluster, as might be expected if the hierarchy can be viewed as an overall ranking. Instead, there was a strong tendency for certain mention forms to be retained, and definite descriptions and single-word names were valid anaphors for most mention types.

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
Fernandes et al. (2012)	72.18	59.17	55.72	62.36	70.51	57.58	53.86	60.65
Björkelund and Kuhn (2014)	73.80	62.00	59.06	64.95	70.72	58.58	55.61	61.63
LIMERIC Baseline	73.66	60.64	57.77	64.02	69.74	55.76	53.34	59.61
+ AR Transitions	74.34	<b>61.81</b>	<b>58.74</b>	<b>64.96</b>	70.33	<b>56.71</b>	<b>54.52</b>	<b>60.52</b>

Table 5.10: Performance of AR Transition prefixing on CoNLL-2012 TEST.

Looking at feature weights between LIMERIC and AR Rankings in Table 5.9, all feature types see an increase in the number of features and a decrease in the average weight. Furthermore, these changes are roughly uniform over the feature classes, with lexicalised features losing the most weight on average and surface cohesion retaining the most. It appears that we have expanded our feature set to learn a roughly equivalent model. Given the attractiveness of compact models, we interpret this result as evidence against using AR Rank models.

### 5.4.2 Benchmarking

We benchmark the performance of LIMERIC with this new prefixing strategy by comparing against our LIMERIC baseline, as well as the two strongest systems from Chapter 4. Compared to LIMERIC, introducing AR Transitions features yields a 0.94% and 0.91% CoNLL score gain on the gold and automatic preprocessing settings, respectively. That is, despite being simple to extract from only surface form information, AR Transitions are a powerful feature because they allow us to improve LIMERIC to within state-of-the-art performance using gold preprocessing, and Fernandes et al. (2012) performance using automatic preprocessing.

To assess whether this performance increase represents a significant improvement, we use the bootstrap re-sampling sign test with 10,000 re-samples. Table 7.11 shows where improvements are the significant with respect to the LIMERIC baseline using bold face for p-values  $< 0.01$  and italics for the standard  $p < 0.05$ . These two thresholds

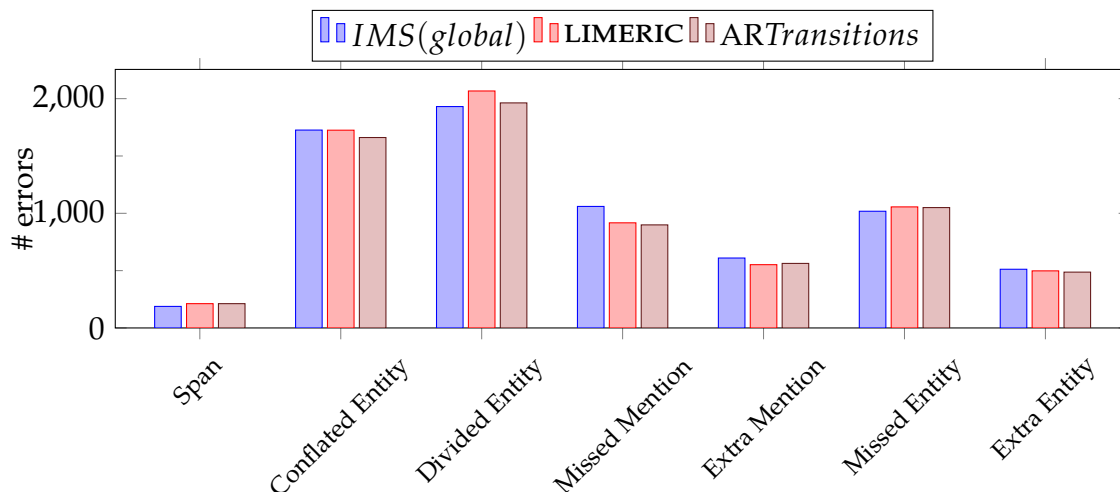


Figure 5.2: Errors made by *AR Transitions* model compared to our *LIMERIC* baseline and *IMS* on CoNLL-2012 TEST using gold preprocessing.

are tested since the three models are not independent, meaning we would expect to see relatively high confidence values for relatively small gains in score (see Berg-Kirkpatrick et al., 2012, for a study).

Compared to the improved *LIMERIC* baseline, our  $B^3$ , CEAFF, and CoNLL scores are all significantly improved on both shared task settings. Interestingly, recall gains are larger than precision gains on the link-based MUC and  $B^3$  metrics. We therefore infer that our significant improvements on CEAFF, which indicate that we are reporting closer to the correct *number* of entities, derive from adding more links between coreferential mentions.

### 5.4.3 Error Analysis

Figures 5.2 and 5.3 show the errors made by the *AR Transition* model on CoNLL-2012 TEST. Analysis of these errors made by the system is consistent with the above interpretation of standard evaluation metrics.

Comparing *AR Transitions* against *LIMERIC* in Figure 5.2, we see that our gains are in the delineation of clusters: the biggest changes are that we reduce the number of

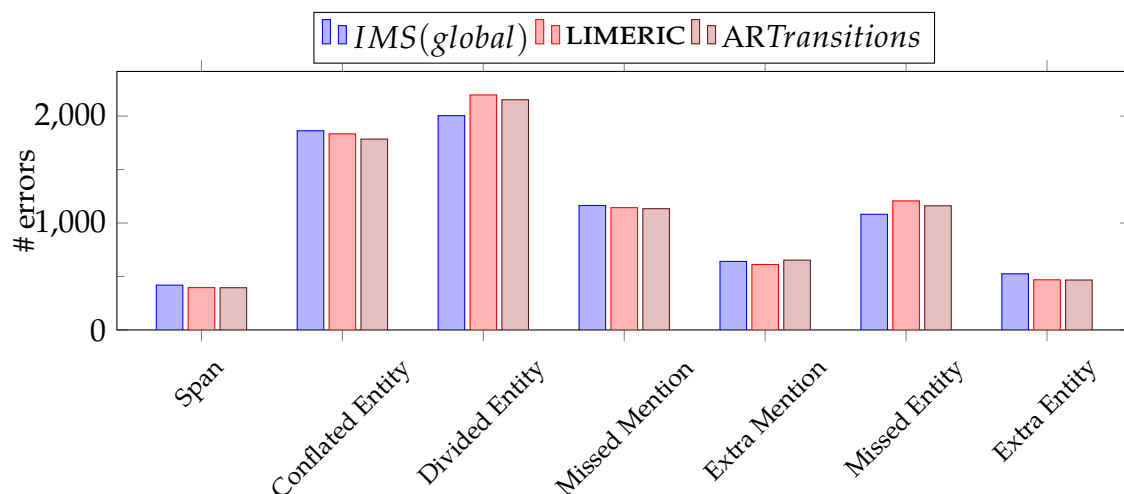


Figure 5.3: Errors made by *AR Transitions* model compared to our *LIMERIC* baseline and *IMS* on CoNLL-2012 TEST using automatic preprocessing.

conflated and divided entity errors. By correcting these errors, which were flagged as problem cases for *LIMERIC* in the previous chapter, we predict closer to the correct number of clusters, hence our improvement in *CEAFE*. Indeed, comparing against the *IMS* system, *LIMERIC* now has a noticeable edge on the *IMS* system, while making a similar number of divided entity errors.

On automatic preprocessing in Figure 5.3, we see similar changes. Both conflated and divided entity errors decrease when we introduction *AR Transitions*. Unfortunately, we continue make more divided entity errors than *IMS* does. The number of times we miss entities also decreases, consistent with our improvements on *CEAFE*, but again does not drop enough to achieve *IMS* performance.

## 5.5 Summary

In this chapter, we have extended *LIMERIC*'s discourse model using the Accessibility hierarchy, a key explanatory mechanism of Ariel's (2001) Accessibility theory. To do this, we devised a mapping of the hierarchy, originally formulated for spoken

Hebrew, to the English text documents in OntoNotes 5.0 and used this mapping in an analysis of the discourse patterns in OntoNotes. We found that the hierarchy's fine-grained classification scheme was useful for understanding the data, and indeed highlighted trends not expressible in a coarser-grained typology, but that there was only limited support for its proposed ordering. We described nuances in reference expression usage in terms of the functions of mentions, for instance that nominals can both indicate coherence by making anaphoric reference, but can also introduce entities and constitute singleton clusters of tangential concepts. We suggest that this multitude discourse functions is a factor in why we don't see uniform transition toward increased accessibility. Future work could investigate whether classifying mentions by their function, rather than their form, affords improvement.

We then grounded this analysis by using our mapping to define discourse transition prefixes. Mirroring the results of our corpus analysis, prefixes based on the hierarchy's classes performed strongly while those based on the hierarchy's ordering made the model sparser but not substantially better than that of LIMERIC. Using *AR* transition prefixes, we significantly boost our performance on CoNLL-2012 and show that this is from reducing our two largest sources of error, conflated and divided entity errors.

In the next chapter, we consider how another of the explanatory factors of Ariel (2001), competition, or, more generally, mutual information, can be used to extend the state of the art for coreference resolution using LIMERIC.

## 6 Mutually Informative Features

*Work described in this chapter forms part of the conference paper Kellie Webster and James R Curran. 2014. Limited memory incremental coreference resolution. In Proceedings of the 25th International Conference on Computational Linguistics, pages 2129–2139.*

This chapter explores the impact of mutual information, which can be viewed as an extension of Ariel’s (2001) competition that allow us to model the human ability to use multiple pieces of evidence simultaneously to resolve coreference. Specifically, we consider the mutual information between coreference indicators at two levels. First, we study the association of features extracted on a given classification instance, to understand how co-occurring features are meaningful when considered as pairs. We find associations between features encoding cohesion and those encoding proximity and topicality, consistent with the argument in cognitive theories that cohesion is insufficient for modelling coreference. Second, we study how the features extracted for one candidate resolution of a mention can influence those of the alternative candidates. Despite being aligned with the motivation of competition learning, we find this information useful for mediating anaphoricity determination.

To encode mutual information, we develop a series of secondary features and implement competition learning in our framework. We find gains from both which are complementary, adding to our CoNLL-2012 scores of 65.29% and 61.13% using gold and automatic preprocessing. These scores are either better or not significantly different from those of Björkelund and Kuhn (2014).

## 6.1 Motivation

While our high-level motivation for this chapter is to model competition, we extend the notion substantially from Ariel (2001). In this way, we build a finer-grained understanding of the factors which contend with one another in resolving coreference, one which aligns with the developing arguments of feature non-independence from Chapter 3. We see this direction as fruitful for two further reasons. Firstly, we have seen that coreference is a complex phenomenon that humans resolve by considering evidence from multiple indicators simultaneously. We would like to build this competency into LIMERIC. Also, given the rise of neural networks, whose strength comes in part from modelling intricate interactions, and the difficulty interpreting their models, we expect the analysis in this chapter to be useful for motivating future work in this space.

### 6.1.1 Antecedent Competition

Accessibility theory (Ariel, 2001) discusses the impact of competition in antecedent selection on reference resolution: when there are multiple compatible resolutions of a mention competing, the salience of each diminishes and this necessitates the use of a lower accessibility mention type. This is equivalent to saying that a more informative mention should be used when there is potential ambiguity about its referent. While we agree that competition impacts coreference resolution, we feel that its formalisation in Accessibility theory fails to capture some important insight.

Firstly, it is not clear what the grounding for having the salience of competing resolutions diminish is, though it does predict that an informative mention type should be used to address ambiguity. However, consistent with Versley (2008) and Recasens et al. (2011) (cf. Chapter 3), we feel that ambiguity can exist at many levels, not just at the high-level choice between entity clusters. We therefore break with Ariel to model competition at the level of *coreference indicators*, rather than the entities themselves. Our implementation of competition learning examines whether the relative salience of

competing candidates impacts resolution, and whether this depends on which feature the candidates match on.

We make a second break from Ariel and study the larger space of *mutual information* between features, in order to capture richer interactions than just direct rivalry. Introducing secondary features allow us to learn that pairs of features are informative when considered together and that these pairings can equally contend with one another, as well as standard coreference indicators, in resolving coreference. In our final system with secondary features and competition learning, we examine whether feature pairs and candidate salience interact in resolving ambiguous reference.

Our two proposed breaks with Ariel fit with arguments of feature non-independence in that both suggest benefit from modelling the interaction between features.

### 6.1.2 Feature Non-Independence

Each of the features implemented in Chapter 4 measure the compatibility of a mention and candidate antecedent cluster in one dimension of our coreference model. Weights for these features sum together independently to give the model's prediction for their likelihood of coreference. However, Björkelund and Farkas (2012) observed that secondary features which conjoin two features were vital to their system's competitive performance. This observation suggests that coreference features are not independent of one another, but rather inter-dependent, and that allowing weights to reflect these inter-dependencies can improve our model. We indeed observe feature associations in this chapter, finding, for instance, that a conjoined feature over NER match and sentence distance improves LIMERIC performance, since NER match is more informative when mentions are in adjacent sentences than when they are in the same sentence.

We saw in Chapter 3 that competition learning approaches to coreference also challenged the assumption of independence between features. In showing that rankers improve modelling, these approaches demonstrated that features generated for the available candidate resolutions of a mention were inter-dependent. For example, we

would expect coreferring a mention and a NER-mismatched cluster, which is often satisfactory, to be less favourable if there exists an NER-match alternative.

In this chapter, we assess the applicability of both methods for enriching our model with feature mutual information. Secondary features are implemented and tested both with and without the simultaneous introduction of competition learning to understand the benefit of each approach to exploiting mutually informative features.

## 6.2 Secondary Features

In this section, we design and test a series of secondary features in the LIMERIC framework. Due to the already large size of LIMERIC's feature set, it is necessary to hand select which feature conjunctions we test, rather than attempting an exhaustive search. We use Chi-Squared ( $\chi^2$ ) association statistics to understand patterns in feature co-occurrence on classification instances in OntoNotes 5 and propose secondary features based on this analysis. That is, this section concerns our first level of study, between features extracted on a given classification instance.

Our approach has the advantage of allowing us to discover interesting trends in reference expression usage, and we discuss the association between features capturing cohesion and those capturing proximity and topicality. We discover secondary patterns in coreference indicators, and leverage these to affect an improvement of 0.35% and 0.37% on CoNLL-2012 DEV. While this gain is not as strong as might be expected based on the Björkelund and Farkas (2012) result, these experiments allow us to analyse trouble cases for our system.

### 6.2.1 Association Statistics

Statistical tests which assess whether two events are dependent do so by attributing their co-occurrence to either (1) a null hypothesis of chance coincidence, or (2) an alternative hypothesis of dependence. To do this, they compare the probability of their

co-occurring against the probabilities of each in isolation. Considering the contingency table in Table 6.1, if the events  $x$  and  $y$  are dependent, we would expect the probability of their co-occurrence,  $p(x, y)$ , to be large compared to  $p(\neg x, y)$  and  $p(x, \neg y)$ , when one event occurs but not the other.

		Feature $x$	
		True	False
Feature $y$	True	$p(x, y)$	$p(\neg x, y)$
	False	$p(x, \neg y)$	$p(\neg x, \neg y)$

Table 6.1: Matrix of outcomes over two possible feature extractions.

In our case, we would like to understand whether two feature extractions are dependent. Considering that our data is binary (a feature is either extracted or it is not), with no apparent base distribution, two tests we could use are Chi-Squared ( $\chi^2$ ) and Pointwise Mutual Information (PMI). Correlation measurements such as Pearson's (1895) or Spearman's (1904) coefficients are unsuitable since our variables are not continuous or orderable, and co-occurrence statistics are less informative than association statistics because they do not account for the expected distribution with respect to non-co-occurrence events.

Both  $\chi^2$  and PMI define test statistics whose increase in magnitude indicates an increasing degree of association between a pair of variables. However, PMI suffers from being simultaneously a function of dependence and entropy, becoming unstable for low frequency events. We therefore choose  $\chi^2$  as our test statistic. Below we discuss calculation and interpretation of  $\chi^2$ , as well as its limitations.

### Test Statistic

The  $\chi^2$  test statistic is given by the following equation, in which  $N$  is the total number of feature extraction events, and  $p(x)$  and  $p(y)$  the overall probability of  $x$  and  $y$ , i.e. both its co-occurrence with its pair, as well as in isolation. This equation is derived

by comparing the observed probabilities of the events in a contingency table against expected values for these probabilities, calculated by assuming independence.

$$\chi^2 = N \cdot \frac{p(x, y)p(\neg x, \neg y) - p(\neg x, y)p(x, \neg y)}{p(x)p(y)p(\neg x)p(\neg y)}$$

The  $\chi^2$  test statistic is typically interpreted using a  $\chi^2$  table, which relates a series of confidence levels (e.g.  $p < 0.05$ , which we use in this chapter) and the degrees of freedom in the test (1 in the case of a feature pair) with a threshold  $\chi^2$  value. If the  $\chi^2$  test statistic is greater than this threshold value, the result is statistically significant in that it would be expected to occur by random co-incidence less than 5% of the time.

The scale and significance thresholds of  $\chi^2$  are known to be affected by dataset sparsity and size. On the one hand, these shortcomings are not overly problematic for this study since we will only be using the magnitude of our  $\chi^2$  statistics to indicate the relative *degree* of dependence between different feature pairs, rather than testing for strict statistical significance. However, we only report on pairings indicated to be significant, and disregard results from contingency tables which contain an expected frequency of less than five, which is a standard approach to limiting the impact of data sparsity (Mooney and Jolliffe, 2003).

## Calculation

In this study, we learn which feature pairs are mutually dependent on a given classification instance (i.e. mention-cluster comparison) by measuring their frequency in processing OntoNotes data and applying the  $\chi^2$  test statistic. That is, we will observe feature probability empirically, according to the relative frequency  $p(x) = \frac{\text{frequency}(x)}{N}$ .

We extract  $\chi^2$  over CoNLL-2012 DEV by processing the dataset with LIMERIC using our development AR Transitions model from the last chapter, trained on TRAIN only. Each comparison generates  $n$  features, and each of these contributes to the frequency tally for that feature and each of the  ${}^nC_2$  pairs adds to the frequency tally of that pair.  $N$  is kept as a running tally throughout this process. We do not consider prefixed features

in this analysis since doing so would inflate the number of feature pairs enormously, and potentially cloud associations that do exist in the data.

### Variants

To better quantify the problem, we keep parallel tallies to calculate three different variants of our test statistic, namely *all*, *div*, and *con*. Specifically, any feature set extracted on an instance consistent with the gold answer key always contributes to *all* test statistics. In this way, *all* statistics indicate what features, when taken together, are associated with coreference.

In the case where the prediction is incorrect, it may either cause an entity cluster to divide (when the prediction is falsely new) or conflate (when the prediction is falsely anaphoric or wrong link). In these error cases, the extracted feature set also contributes toward the *div* and *con* test statistics, respectively. Therefore, *div* statistics will tell us which feature associations tends to occur in error cases in which we miss a coreference relationship, and *con* which of these occur in cases we propose spurious coreference relationships.

Feature pairs with significant test statistics at  $p = 0.05$  are interpreted as dependent. We interpret an indicated dependence of a feature pair on *all* as a pairing which reliably indicates coreference. Similarly, significant pairings on the *div* statistic also indicate coreference, but are pairings which are not captured by LIMERIC's model. In contrast, significant *con* pairings highlight pairs which appear to indicate coreference, but are instead distractor mentions.

### 6.2.2 Observed Associations

We now give our qualitative impressions of the large volume of feature  $\chi^2$  statistics data, using the statistics as quantitative grounding of trends we highlight. We structure our discussion around our cohesion feature classes considering, for each, their association with those of proximity and topicality. For surface form cohesion, we

additionally consider the association of head match with lexicalised features, since we observe interesting dependencies in this space. All reported  $\chi^2$  values are significant values on *all*. Significant pairings on the *div* and *con* statistics are indicated with the superscripts <sup>d</sup> and <sup>c</sup>, respectively.

## Surface Cohesion

**Head Match vs. Lexicalised Features** The association of head match features with lexicalised features on *all* statistics tells us which head words are likely to participate in cohesion-mediated coreference. Table 6.2 tabulates association statistics for different values of our head match feature over the POS tag<sup>1</sup> of the head word. We remember that these surface form cohesion features take values up to 5 to reflect the number of mentions in an entity cluster with the same head as the current mention.

Head	NNP	NNPS	NN	NNS
1	75620		18347 <sup>d</sup>	3872
2	36736		4258	668
3	20580		1815	
4			888	
5	27758		730	126

Table 6.2:  $\chi^2$  values for different pairings of head match and head POS tag features.

Comparing the columns, we can see that  $\chi^2$  values are larger for head match where the head word has POS tag NNP than when it has POS tag NN, and that these values are in turn higher than those for POS tag NNS. We interpret these statistics to indicate that head match on proper names is a more reliable indicator of coreference than head match on nominals, and that head match on singular mentions is more reliable than on plural mentions. This makes sense: proper names pick out their referent with less ambiguity than common nouns do (e.g. ‘*Barack Obama*’ vs. ‘*the president*’), and groups of

<sup>1</sup>OntoNotes uses Penn TreeBank POS tags ([https://www.ling.upenn.edu/courses/Fall\\_2003/ling001/penn\\_treebank\\_pos.html](https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html)) in which NNP signifies a singular proper noun, NNPS a plural proper noun, NN a singular common noun, and NNS a plural common noun

Head Word	Coref	Total	%
son	192	3358	5.7
market	796	1662	47.9
time	66	1584	4.2
law	1021	1574	64.9
man	647	1428	45.4
one	49	1277	3.8
father	167	958	17.4
government	487	937	52.0
city	594	909	65.3
world	549	880	62.4

Table 6.3: Proportion of head matched nominal mention pairs which are coreferential.

entities indicated by plural forms need to match exactly in composition to be annotated as coreferential (e.g. *‘the protesters’* can equally refer to a subset of a given group as the group itself, given the correct context). Interestingly, no head match features are significantly associated with mentions headed by plural proper names, with POS tag NNPS. Perhaps head match on plural proper names is particularly sensitive to this set-subset problem, given that proper names tend to pick out their referent with little ambiguity (i.e. we can be more sure when similar groups are not identical).

Considering the case of head matched nominal mentions further, Table 6.3 gives coreferentiality statistics for the ten most common words occurring as the head word of a singular nominal mention. Specifically, all pairings of automatically extracted mentions are generated and the number of times these match on head word are tallied by head word. A second tally captures whether the aligned gold mentions, if they exist, are annotated as belonging to the same entity cluster. These tallies respectively give the counts in the third and second columns, while the fourth simply expresses these statistics as a percentage.

We can see that the likelihood that head term matched mentions are coreferential is distributed between 0 and 65.3%: although a highly trusted feature, head match

Sentences	exact	relaxed	head	Depth	exact	relaxed	head
0			99932	top			
1		54633	132031 <sup>d</sup>	upper	26756	29438	147806 <sup>d</sup>
2				lower			
				bottom			214402 <sup>d</sup>

(a) Sentence distance

(b) Stack depth

Table 6.4:  $\chi^2$  statistics for different pairings of surface form cohesion and proximity features.

does not uniformly indicate coreference. Indeed, for the examples given, head match indicates coreference at most 65% of the time. Manual analysis of non-coreferential head-matched mentions show that there are many factors at play, including the specificity of a mention's referent (e.g. *'the president'* is markable when it refers to a particular individual, but not when it refers generically to the role), the presence of restrictive modification (e.g. *'the junk market'* cf. *'the stock market'*), as well as genre preferences (head match is more reliable among terms associated with the Biblical domain than those of the financial). Perhaps due to these factors, LIMERIC learns to be overly conservative, and makes divided entity errors on nominal comparisons.

**Surface Form Cohesion vs. Proximity** Table 6.4 gives the  $\chi^2$  values for pairs of our various surface-level cohesion features with different values of the sentence distance and depth features. These statistics allow us to read the scope over which cohesion features reliably indicate coreference. Count based features (i.e. exact and relaxed string match, and head match) are for a value of 1 (adjacent sentences), which is the only value consistently significant for the pairs tested. No feature pairs for sentence distances greater than 2 are significant, and the rise in  $\chi^2$  in the bottom depth value of the stack could be an effect of how the feature is defined, since this zone takes in all depths greater than 9.

Overall, we can see that positive evaluation of cohesion features is most strongly associated with close proximity when there is a coreference relationship: these two factors indeed conspire as Accessibility theory permits them to. Specifically, the preferred contexts for surface-form cohesion is when mentions are in adjacent sentences, or when entity clusters are in the upper zone of the stack. While close-range string match is not observed on pairs with sentence distance features, it is with depth features. We suggest that this further supports the validity of our cognitive depth measure as an indicator for reference expression distance.

**Topicality vs. Surface Form Cohesion** Looking at the association between surface-form cohesion features and our topicality indicator, cluster length, only one feature pair is significant. Head match with value 1 (i.e. the mention shares its head with one mention in the candidate cluster) has a  $\chi^2$  value of 402804 on our DEV dataset with cluster length of 1, and this pair is also significant on the *div* statistics. As well as on nominals and in adjacent sentences, head match is often missed on comparisons involving discourse singletons. Given that all clusters grow incrementally from singletons, this conservativeness is important to address.

### Attribute Cohesion

In the following exploration of the association of attribute cohesion, we find that many reviewed pairs are significant on the *div* statistic as well as *all*. This presumably reflects attribute cohesion being a weaker indicator of coreference than surface-form cohesion, and means that the associations discovered would seem to have good promise for addressing LIMERIC's divided entity errors seen in the previous chapters.

**Attribute Match Pairs** Tables 6.5a and 6.5b tabulate  $\chi^2$  statistics over pairs of attribute match features. Table 6.5a gives  $\chi^2$  when both features in the pair are positive (i.e. both attributes have matches) while Table 6.5b gives  $\chi^2$  when one attribute is positive (horizontal) and one is negative (vertical).

	NER	gender	number
gender	140474		
number	322613 <sup>d</sup>	<b>663526<sup>d</sup></b>	
animacy	511968 <sup>d</sup>	<b>553513<sup>d</sup></b>	<b>1361641<sup>dc</sup></b>

(a) Positive attribute match features

True \ False	True	NER	gender	number	animacy
	False	NER	gender	number	animacy
NER			<b>401641<sup>d</sup></b>	<b>1103286<sup>dc</sup></b>	<b>1105318<sup>dc</sup></b>
gender		338144 <sup>d</sup>		787887 <sup>dc</sup>	1061948 <sup>dc</sup>
number		175338			255661 <sup>d</sup>
animacy				65804 <sup>d</sup>	

(b) Positive and negative attribute match features

Table 6.5:  $\chi^2$  statistics for pairs of attribute match features.

Looking at Table 6.5a, we can see that all pairings of positive attribute match feature are associated on the *all* statistic. This is reassuring, since all attributes are expected to indicate coreference. We can also see that NER behaves differently to the other attributes in that associations of positive non-NER attribute matches have higher  $\chi^2$  values. We suggest this might be due to the raw number of matches seen given the dependencies between different attributes: number matches are the broadest filter, with number matched mentions possibly animacy or gender matched, and NER is the finest-grained filter with animacy and gender matched mentions possibly NER matched. For instance, ‘*the spokesman*’ and ‘*the spokeswoman*’ are number matched (singular) but not gender matched, and ‘*the company*’ and ‘*the stock*’ are number (singular), animacy (inanimate) and gender (neuter) matched, but not NER matched. In this way, we would expect number, animacy, and gender to co-occur more frequently in coreference data.

Looking now at Table 6.5b, the strongest associations on the *all* statistic are for attribute pairs without an NER match. This is consistent with our observation that NER match will occur less frequently than matches on the other attributes. Addition-

Sentences	NER	gender	animacy	number
0	27880	<b>133962</b>	<b>480693<sup>d</sup></b>	<b>448027<sup>d</sup></b>
1	176350 <sup>d</sup>	<b>233191</b>	<b>584613<sup>d</sup></b>	<b>510626<sup>d</sup></b>
2			168930 <sup>d</sup>	137129 <sup>d</sup>
3			87485	77344

(a) Sentence distance features

Stack	NER	gender	animacy	number
top			188656 <sup>d</sup>	175840 <sup>d</sup>
upper	83062 <sup>d</sup>	<b>284792<sup>d</sup></b>	<b>811060<sup>d</sup></b>	<b>746818<sup>d</sup></b>
lower			200402 <sup>d</sup>	165137 <sup>d</sup>
bottom	439781 <sup>d</sup>	132008	411352 <sup>d</sup>	330605 <sup>d</sup>

(b) Stack depth features

Table 6.6:  $\chi^2$  statistics for different pairings of attribute cohesion and proximity features.

ally, these pairings also correspond to both divided and conflated entity errors. This also makes sense: the evidence from the broader attributes is not strong enough for LIMERIC to make the correct decision. On the other hand, cases without animacy or number match are either not significantly or only weakly associated with coreference on *all*: these attributes are, to some extent, necessary but not sufficient to determine a coreference relationship.

**Attribute Cohesion vs. Distance** Tables 6.6a and 6.6b give the  $\chi^2$  values for pairs of positive attribute cohesion features with our proximity features. Comparing against Table 6.4, we find that attribute matches can operate over longer ranges than surface form matches, with both the upper and bottom depths of the stack associated with attribute-mediated coreference and animacy and number match significantly associated with sentence distances up to three.

Cluster	NER	Gender	Animacy	Number
1	167389 <sup>d</sup>	106658 <sup>d</sup>	393294 <sup>dc</sup>	409618 <sup>d</sup>
2		74161	241624 <sup>d</sup>	215218 <sup>d</sup>
3			165668	139919
4			116455 <sup>d</sup>	99566
5			90513	

Table 6.7:  $\chi^2$  statistics for the association of topicality and our various cohesion features.

Interestingly, attribute match is more strongly indicative of coreference when mentions are in adjacent sentences than when they are in the same sentence and this preference is particularly strong for NER and, to a lesser extent, gender match. Examining non-coreferential intra-sentence instances manually reveals that they correspond to entities of the same type related by a predicate, e.g. two people reported as participating in the one event.

Inspecting instances of animacy or number agreement in the same and adjacent sentence contexts shows that these cases are very hard to resolve. Instances in adjacent sentences are mostly cases where a definite pronoun needs to be resolved to a proper name or description based on inference about the entities from their verb frame. For example, in the following sentence, coreference between the indicated mentions is cued in the fact that ships are unloaded from carriers, thereby making ‘*the ship*’ the best antecedent for ‘*it*’. Such challenging cases of coreference are the target of the Winograd Schema Challenge (Rahman and Ng, 2012), which we consider in the next chapter.

It will be welded to **the ship** before **it** is unloaded from the carrier.

**Attribute Cohesion vs. Topicality** Table 6.7 gives the  $\chi^2$  values for pairs of attribute cohesion and cluster length features. Where we saw that surface form features were not strongly associated with any particular length of cluster, attribute match, particularly

Class	Features
Surface Cohesion + Depth	head match (T/F) + depth (raw) relaxed string match (T/F) + depth (raw)
Head Match	head match (T/F) + both specific head match (T/F) + cluster length head match (T/F) + document genre head match (T/F) + head word
Attribute Cohesion Pairs	NER + animacy agreement NER + gender agreement NER + number agreement animacy + gender agreement animacy + number agreement gender + number agreement
Attribute Cohesion + Distance	NER agreement + sentence distance animacy agreement + sentence distance gender agreement + sentence distance number agreement + sentence distance
Attribute Cohesion + Topicality	NER agreement + cluster length animacy agreement + cluster length gender agreement + cluster length number agreement + cluster length

Table 6.8: Secondary feature set of conjunctive features.

animacy and number match, is. We suggest this is related to larger clusters tending to comprise chains of pronouns, for which attribute information can be reliably assigned.

We can also see that each of our attribute cohesion features is most reliably associated with single-mention clusters on both the *all* and *div* statistics. This is probably due to all clusters being built incrementally from single- to multi-mention clusters.

### 6.2.3 Secondary Features

Using the above analysis, we design and test secondary, conjunctive features in LIMERIC. Table 6.8 summarises the features we introduce and Table 6.9 their impact on system performance.

We can see that, while secondary features improve system performance on both the gold and automatic preprocessing settings, their impact on our strong baseline is modest compared to what might be expected given the Björkelund and Farkas result. The gains follow the same trends in both gold and automatic settings, but the impact of individual feature classes tends to be greater using automatic preprocessing. Unfortunately, the gains from single feature classes is not highly additive, despite being

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
AR Baseline	73.80	61.98	60.26	65.35	69.55	56.99	55.35	60.63
Surface + Depth	73.76	61.76	60.04	65.19	69.67	56.93	55.29	60.63
Head Match	74.14	62.29	60.48	65.64	69.75	57.29	55.60	60.88
Attribute Pairs	73.97	62.12	60.27	65.45	69.69	57.25	55.66	60.87
Attribute + Distance	73.84	62.19	60.18	65.40	69.73	57.40	55.56	60.90
Attribute + Topicality	73.79	61.95	60.06	65.27	69.77	57.37	55.69	60.94
All (no Surface + Depth)	74.22	62.42	60.47	65.70	69.88	57.52	55.60	61.00

Table 6.9: Performance of secondary features on CoNLL-2012 DEV.

based on different aspects of the above analysis. This is particularly the case on the automatic setting where each class except surface cohesion + depth affects at least a 0.26% performance boost, but together add 0.37% to system performance. While we would not expect perfect complementarity, it could also be that the large number of features now in our model is approaching the bounds of what is learnable given the amount of training data in OntoNotes.

### Surface Cohesion + Depth

The above analysis indicated that surface cohesion matches were associated with depth in the stack, but not the number of sentences between mentions. We therefore conjoin depth with each of relaxed string match and head match features. Unfortunately, this feature class performs poorly and to understand why, we tabulate the weight of each of the unprefixed versions of these features in Table 6.10, as well as the CoNLL score of a model trained with just relaxed string or head match.

For good performance, we would expect a large margin between the weights of surface form match (True) and no surface form match (False). We indeed see this at all depths for the relaxed string match secondary features, and for non-top depths of head match conjunctions. Indeed, at the top depth, *not* having a head match is learned to be a better indicator of coreference than having a match.

	String Match		Head Match	
	True	False	True	False
Gold	65.27		65.43	
top	0.25	-0.04	0.20	0.36
upper	0.56	0.02	0.74	0.18
lower	0.53	0.10	0.74	0.15
bottom	0.60	0.01	0.66	0.04

Table 6.10: Weights of unprefix surface form cohesion and depth secondary features in the Surface + Depth model.

We note in particular that this margin is wider in the important upper depth (which the above analysis associated with coreference) for our head match secondary feature; this favourable outcome bears out in the head match secondary feature model having stronger performance than our relaxed string match model. Despite the satisfactory performance of the head match secondary feature, we opt to exclude this as a feature class in the work following.

### Head Match

Within surface cohesion, head match was an interesting target for investigation. While a highly trusted feature in LIMERIC’s model, head match was shown above to not be uniformly trustworthy, but instead its association with coreference is a function of the head word itself, its part of speech, the specificity of the mention, as well as the genre of its document.

The Head Match feature class comprises four secondary features conjoining head match with mention specificity, cluster length, document genre, and head word. This is the strongest feature class on the gold setting and performs similarly well on automatic preprocessing.

Genre	Weight
Telephone conversation	0.45
Bible text	0.43
Broadcast conversation	0.43
Broadcast news	0.41
Newswire	0.07
Magazine text	-0.00
Web text	-0.15

Table 6.11: Weights of unprefix features conjoining document genre and head word match, in the Head Match model.

**Head Match + Both Specific** OntoNotes guidelines that markable units should make specific reference (or be coreferential with another unit which does). We therefore conjoin head match with an indicator of whether the head matched mentions (the closest head matched antecedent, with the current mention) are both started by either the definite determiner or a possessive or demonstrative pronoun.

**Head Match + Cluster Length** Head match was the only surface cohesion feature associated with topicality on the *all* statistic, preferably applying between a mention and a new (singleton) discourse entity. We therefore encode a feature conjoining head match with the cluster length, capped at size 3.

**Head Match + Document Genre** To allow the preference for anaphoric head match according to the six genres represented in OntoNotes, we conjoin positive head match features with document genre, as indicated from OntoNotes document names. Given that genre preference was merely a qualitative impression above and not quantified, we give the weight assigned to the unprefix features in this class in Table 6.11. These weights show that document genre indeed is used by the learner, with head match importance being boosted in four out of the seven OntoNotes genres.

Interestingly, despite the genre’s importance in the development of shared task conditions, head match is not learned to be a reliable feature in newswire. Reviewing newswire documents, we see that for many common head words (such as markets, economies, and companies), multiple entities will be referred to using the word as a nominal head. In these cases, the different entity references are indicated, in the first two instances, by modifiers and by anaphoric reference to previous organisation names in the latter.

**Head Match + Head Word** The association that head match can be trusted more between proper names than nominal mentions is already captured in our discourse transition prefixes. To instead introduce the finer-grained patterns we saw in Table 6.3, we introduce a conjunctive feature between positive head matches and the head word of the mention, or its POS tag if the word occurs fewer than 50 times in the training data.

### Attribute Match Pairs

For each of the seven pairs of the four attributes gender, number, animacy, and semantic class, we conjoin the agreement features, allowing us to learn that, for instance, number or animacy disagreement in NER matched mentions is not associated with coreference. That is, we re-implement the attribute agreement conjunctions of Culotta et al. (2006). Even on our strong baseline, these simple secondary features perform well. Despite this, we present feature weights in Table 6.12 since it reveals some interesting anomalies when compared to Table 6.5.

We would expect to see higher feature weights learned where  $\chi^2$  association statistics were greater and, overall, this is what we see. However, the weight for gender + animacy match in Table 6.12a is lower than expected given their high association. We suggest this could be because the attribute value determinations for gender and animacy are correlated, and their match is learnable without secondary features. Also,

	NER	gender	number
gender	0.52		
number	0.72	1.21	
animacy	0.97	0.69	1.11

(a) Both positive matches

False \ True	NER	gender	number	animacy
	NER	gender	number	animacy
NER		0.70	0.96	0.64
gender	0.79			0.92
number	0.59			0.50
animacy		0.53	0.56	

(b) Positive and negative matches

Table 6.12: Weights of the unprefixd paired attribute match features on Attribute Pairs model.

the weight for gender agreement + animacy disagreement in Table 6.12b is higher than we expect, which we interpret to mean that animacy mismatch is highly informative when gender matches (i.e. on neuter gendered entities). Finally, the weight for animacy agreement and NER class disagreement is lower than expected. This is unfortunate given that this pair was highly associated for coreference above, and was associated with instances where LIMERIC makes divided and conflated entity errors. Given that we assign NER mismatch in cases where NER cannot be determined, improving the lexical semantic classification of nominal mentions should therefore be a target for future work.

### Attribute Match + Distance

We conjoin each attribute match feature with the distance in sentences between the current mention with a feature indicating whether the current mention and the closest

len	NER	gender	animacy	number
1	0.44	0.47	0.55	0.52
2	0.58	0.56	0.64	0.59
3	0.66	0.67	0.69	0.68

Table 6.13: Weight of the unprefixed features conjoining attribute match with cluster length in our Attribute + Topicality model.

antecedent in the candidate entity cluster are within a sentence distance of two from one another.

### Attribute Match + Topicality

We conjoin each of the attribute agreement features with cluster length, capped at length three. Topicality improves performance on automatic preprocessing, but there is a small drop on the gold setting; given our weak performance on automatic relative to gold preprocessing, we see this result to be an acceptable compromise, particularly given the small magnitude of the drop in CoNLL score on gold.

The feature weights in Table 6.13 show that LIMERIC has learned to trust attributes more for larger clusters. While this is contrary to the modelling in our association statistics, it is intuitively sound given the motivation for cluster-level modelling to improve confidence of cluster-mention comparisons by pooling properties across mentions in an entity cluster.

## 6.3 Feature Competition

In this section, we explore how the features extracted for competing candidate resolutions are mutually informative. In Chapter 3, we saw that such modelling has been useful in competition models of coreference resolution; we extend this here by incorporating candidate salience, approximated by its position in the forest of entities, in competition feature extraction.

We identify two complementary ways in which competition can be implemented: competition in the stack and anaphoricity competition. We find that anaphoricity competition, which exploits competition features on the shift, or discourse-new, classification, thereby mediating anaphoricity determination, is particularly successful. Using automatic preprocessing, these features boost performance on CoNLL-2012 DEV by 0.51% without secondary features and 0.45% against their stronger baseline.

### 6.3.1 Experimental Setup

Our experimental design relies on our discourse entities being stored in a self-ordering data structure according to their relative accessibility. We profiled this forest in Chapter 4 and found that correct choices for antecedent tend to be located near the top of this data structure. Therefore, we expect that if there are multiple compatible entities competing, the most accessible of these should be, on average, the best choice.

We design a series of experiments to model competition between antecedents directly in our feature set, giving the most accessible of a group of compatible entities enhanced prominence compared to lower ranked matches. Implementing this in LIMERIC is straightforward, and we do so by including a key-value store whose keys are feature functions and values are booleans reflecting whether the given function has had a positive value extracted for it thus far in processing. Feature functions which return boolean or integer values are acceptable keys, and values of True or values  $\geq 0$  are taken to indicate compatibility. In this way, it is possible to flag on which candidate a particular feature function is first satisfied and prefer this candidate accordingly.

### 6.3.2 Forms of Competition

The information captured in this key-value store is used to define our two variants of competition, competition in the stack and anaphoricity competition, as well as their combination, full competition.

### Competition in the Stack

In stack competition, each comparison between the current mention and a given entity cluster generates two features, as shown in Figure 6.1. The first feature is the standard feature, while the value of the second reflects whether the cluster is above or below the first compatible entity, as indicated by that feature. The above feature value is intended to inform the learner that despite the negative evaluation, this candidate is more accessible than the best choice. On the other hand, the below feature value captures the diminished prominence of the candidate with respect to what the feature considers to be the best choice.

For the cluster of interest itself, *‘Aden Harbor’* in our example, we experiment with three different feature values for the second feature. The cluster could either be labelled with one of the existing tags above or below since it is indeed in the above zone in which entities have yet to be matched on that feature, and in the below zone since it marks when this feature value should start to be used. Webster and Curran (2014) report on the below-match variant. Table 6.14 shows the performance of these two choices. Using above-match to model feature competition allows us to improve our CoNLL score when using automatic preprocessing and does not compromise our performance on the gold setting. Interestingly, switching to below-match yields a 0.35% drop on the gold setting, with minimal increase on the automatic setting. We interpret these results to indicate that resolution is more sensitive to entities high in the discourse stack (i.e. salient entities) than those lower than the first compatible cluster.

Perhaps the most satisfying solution is to introduce a third feature value, *first*, which labels this uniquely as the best candidate on the given feature function. This variant performs remarkably similar to the below-match variant. We can then infer that the important information provided by our competition features is in identifying the depth at which candidates can start to be considered amenable to coreference.

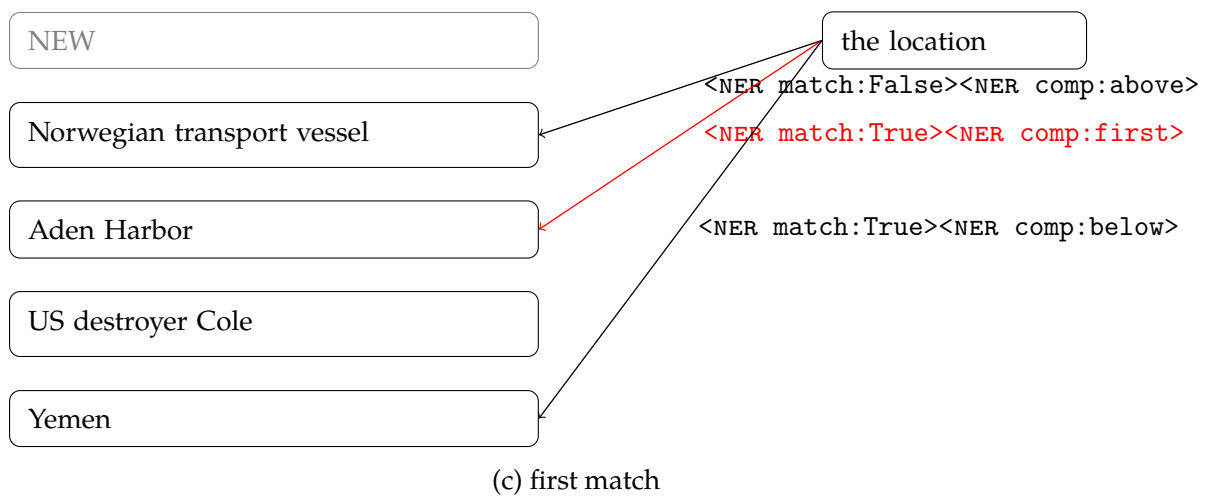
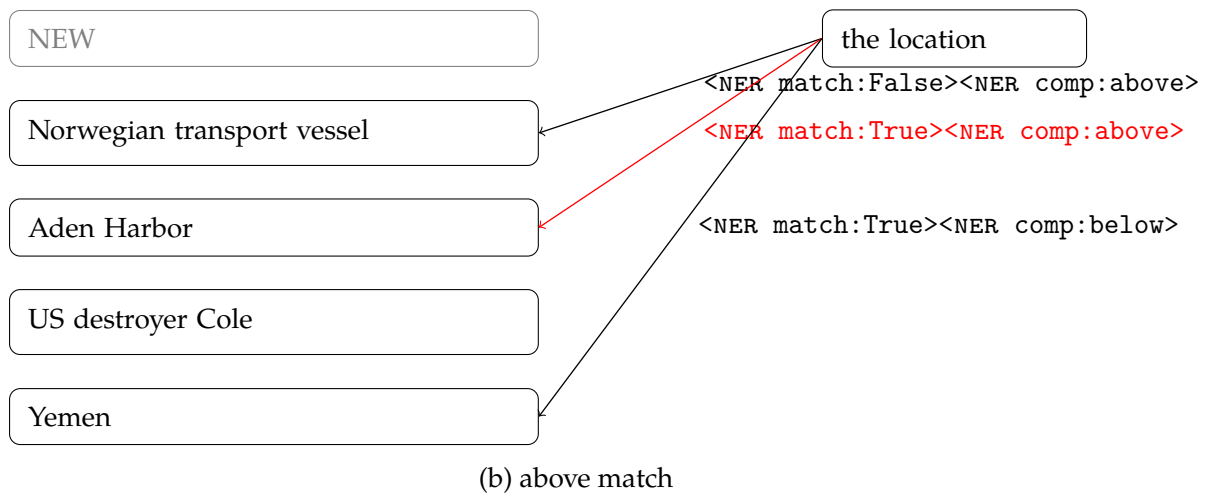
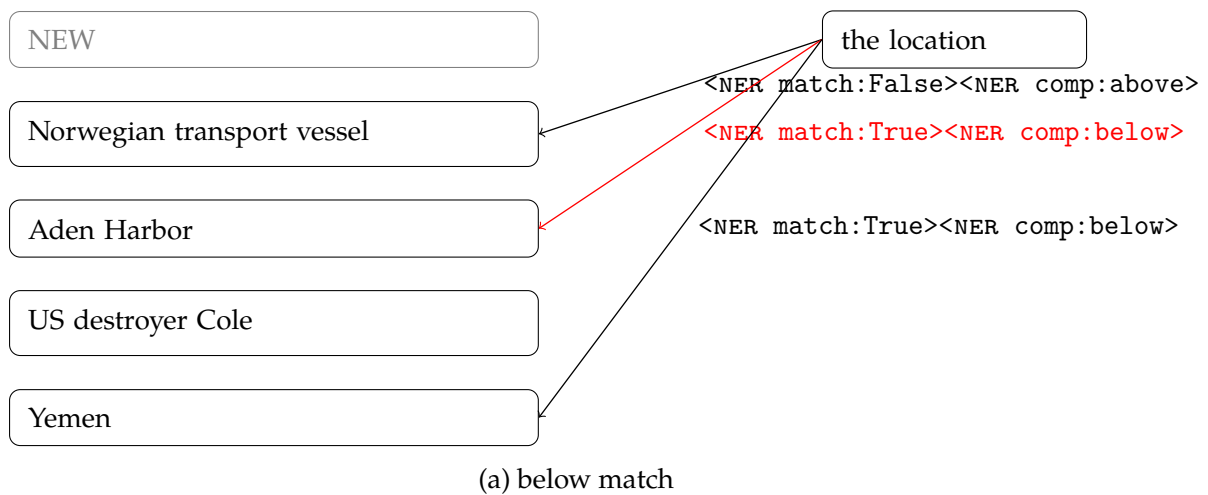


Figure 6.1: Example of stack competition feature extraction.

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
AR Baseline	73.80	61.98	60.26	65.35	69.55	56.99	55.35	60.63
Stack - above match	73.89	62.11	60.14	65.38	69.82	57.29	55.52	60.88
Stack - below match	73.75	61.44	59.80	65.00	69.53	57.07	55.43	60.68
Stack - first match	73.98	61.86	60.33	65.39	69.77	57.18	55.71	60.89
Anaphoricity Competition	73.86	61.85	60.41	65.37	69.85	57.55	56.02	61.14
Full Competition	73.98	62.20	60.52	65.57	69.96	57.71	56.25	61.31

Table 6.14: Performance of competition features on CoNLL-2012 DEV.

### Anaphoricity Competition

LIMERIC’s feature set (cf. Chapter 4) does not include any features specifically designed to capture anaphoricity determination or singleton detection. Instead, for a mention to be labelled anaphoric, its score with a particular entity cluster needs to be greater than the score for shift. Error analysis has shown that LIMERIC is overly conservative in making this decision.

Anaphoricity competition allows a mention’s classification as anaphoric or discourse-new be mediated explicitly in the feature set. Competition is implemented by generating an extra set of features on the shift comparison. Specifically, for all feature functions in our new key-value store, we generate a feature indicating whether it has been satisfied by any candidate in the discourse stack. Such features, for instance, allow the discourse-new comparison to know whether any cluster has an exact string match with the current mention, in which case we would not typically expect the mention to be a discourse singleton, regardless of how well the string-matched entity cluster scores on classification overall.

Table 6.14 shows that anaphoricity competition is successful in improving system performance on automatic preprocessed data, improving CoNLL by 0.51%. This gain on automatic preprocessing is particularly seen on B<sup>3</sup> and CEAFE. On B<sup>3</sup>, recall and precision both increase (by 0.57% and 0.52%, respectively), while on CEAFE the recall

Competition Feature	True	False	Diff.
Indefinite Mention	0.21	5.62	5.41
Shared Sense	0.00	5.83	5.83
Acronym	1.25	4.58	3.33
Possessive Match	1.85	3.98	2.14
Relaxed String Match	2.33	3.51	1.18
String Match	2.52	3.31	0.79
Head Match	2.68	3.15	0.47
Overlap	2.73	3.10	0.37
Words Match	2.79	3.05	0.26
Head Substring	2.88	2.95	0.13
Mention Substring	2.90	2.93	0.03
Mention Length Match	2.96	2.88	-0.08
Head Edit Distance	2.98	2.85	-0.13
Gender Agree	2.49	3.34	0.85
NER Agree	2.88	2.95	0.07
Animacy Agree	2.97	2.86	-0.11
Number Agree	3.06	2.77	-0.29

Table 6.15: Average weight of anaphoricity competition features.

gain (1.11%) far outweighs the precision gain (0.11%). We can infer that anaphoricity competition is effective in recalling entity clusters, whose presence and absence particularly impacts CEAFE.

To explore this further, Table 6.15 tabulates the average weights of unprefixd anaphoricity competition features. Since the True column indicates when there is a compatible cluster in the discourse stack, and False, when there is no match in the stack, we can interpret a high value on False relative to True as evidence that a feature is particularly strong for forcing a coreference relation. Surface form heuristics, in the upper section, and attribute agreements, in the lower, are therefore ordered by the magnitude of this False to True difference.

Our first observation is that, compared to the average feature weights for previous LIMERIC models, the feature weights here are quite high. This means that our learner

trusts these features as reliable indicators of anaphoricity and coreference. Next, we can see that relaxed string match is a more robust indicator of coreference than exact string match. Other surface form cohesion statistics follow expectations, being ordered by how readily they can be satisfied on a particular comparison. This is consistent with work on anaphoricity determination described in Chapter 4 which found that string and head match were powerful features in their models. We note that the robustness of anaphoricity competition to automatic preprocessing makes sense given this importance of surface form competition features since these features do not depend on NER or syntactic analysis.

On attribute cohesion, gender and, to a lesser extent NER, are robust indicators that a coreference relationship exists. As on our above analysis, animacy and number do not provide our learner strong enough evidence to force coreference.

### **Full Competition**

On full competition, we generate stack competition features as we iterate over discourse entities and additionally generate anaphoricity competition features on the discourse-new comparison. Table 6.14 shows that, while neither stack competition nor anaphoricity competition improved performance on gold preprocessing in isolation, full competition affords a 0.22% CoNLL score gain. However, their impact on performance is more prominent on the automatic setting, giving complementary gains which sum to a 0.68% improvement above baseline.

### **Secondary Feature Competition**

Of the secondary features introduced above, all of the attribute pairs features are suitable for competition features in that they are boolean valued. Table 6.16 shows the performance of LIMERIC with feature mutual information encoded with both secondary features and competition. Secondary features exclude Surface + Depth and stack competition uses the ‘first match’ implementation.

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
Second Order Baseline	74.22	62.42	60.47	65.70	69.88	57.52	55.60	61.00
Competition in the Stack	74.37	62.51	60.39	65.76	69.93	57.52	55.79	61.08
Anaphoricity Competition	74.08	62.25	60.89	65.74	70.06	57.92	56.36	61.45
Full Competition	74.35	62.40	60.46	65.74	70.11	57.70	56.03	61.28

Table 6.16: Performance of secondary (AR Transitions) and competition (Mutual Information) features on CoNLL-2012 DEV.

Introducing competition into our stronger secondary feature baseline gives improvements on both settings and for all forms of competition. But where full competition was superior to anaphoricity competition above, the reverse is true here. On gold preprocessing, the two achieve the same CoNLL score, but on the automatic setting, both B<sup>3</sup> and CEAFE are stronger on anaphoricity competition and MUC is little different between the two. We therefore choose this configuration with secondary features and anaphoricity competition as the best implementation of how LIMERIC can leverage the mutual information in feature extractions to enrich its model.

## 6.4 Evaluation

We evaluate the performance of our system enhanced with mutual information features against our models from the previous chapter in using the same setup described there.

### 6.4.1 Benchmarking

In Table 6.17, we can see that the performance of our mutual information system does not change appreciably using gold preprocessing compared to the baseline set using AR Transition prefixes. However, our performance has improved using automatic

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
Fernandes et al. (2012)	72.18	59.17	55.72	62.36	70.51	57.58	53.86	60.65
Björkelund and Kuhn (2014)	73.80	62.00	59.06	64.95	70.72	58.58	55.61	61.63
LIMERIC Baseline	73.66	60.64	57.77	64.02	69.74	55.76	53.34	59.61
+ AR Transitions	74.34	<b>61.81</b>	<b>58.74</b>	<b>64.96</b>	70.33	<b>56.71</b>	<b>54.52</b>	<b>60.52</b>
+ Mutual Information	74.73	61.72	59.43	65.29	70.72	57.40	55.26	<b>61.13</b>

Table 6.17: Performance of secondary and competition features on CoNLL-2012 TEST.

preprocessing to a point where we are strongly competitive with the state of the art set by Björkelund and Kuhn (2014)<sup>2</sup>.

The results of our statistical significance testing reflect these same observations. Specifically, none of the changes using gold preprocessing are significant, even at the permissive level of  $p = 0.05$ . On the other hand, our improvement on CoNLL using automatic preprocessing is significant at the conservative level of  $p = 0.01$ , and the improvements in B<sup>3</sup> and CEAFE are additionally significant at  $p = 0.05$ . Buoyed by these positive results, we further test whether there is any actual difference between us and the apparently stronger IMS system on automatic. We find that it is not statistically better on any metric, with B<sup>3</sup> being the closest call: there is just a 7% probability that chance accounts for IMS outperforming our Mutual Information model.

Our improvement on the automatic setting arises from increases in both recall and precision. On B<sup>3</sup>, precision increases more than recall (0.98% vs. 0.50%), but on CEAFE we get the opposite and recall increases more than precision (0.81% vs. 0.63%). That is, we appear to recall more correct entity clusters by better delineating the bounds of the clusters themselves.

<sup>2</sup>As noted in Chapter 3, the current best reported performance is Wiseman et al. (2015)

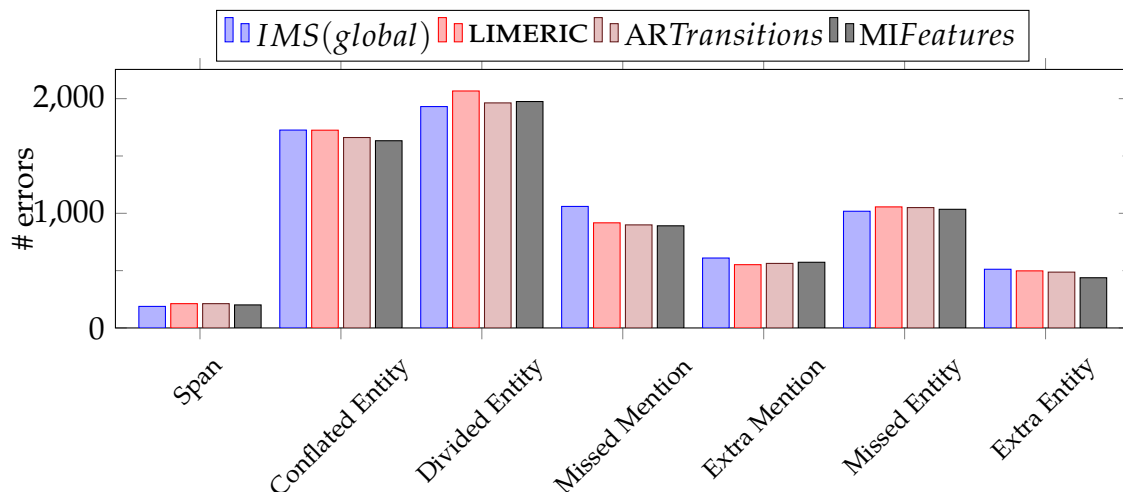


Figure 6.2: Errors made by our Mutual Information model compared to our previous models and IMS on CoNLL-2012 TEST using gold preprocessing.

### 6.4.2 Error Analysis

Figures 6.2 and 6.3 show the error distributions of each of our models against the benchmark Björkelund and Kuhn (2014) system. On both settings, we see a further drop on conflated entity errors. Using gold preprocessing affects a drop in extra entity errors while automatic preprocessing improves the trouble case on divided entity errors. Given that the impact of anaphoricity determination was largely limited to the automatic setting, we expect the changes on gold to largely be affected by our novel secondary features. These, it would appear, act to rule out spurious links, perhaps from more accurate modelling of when cohesion should be trusted. On automatic preprocessing, our model is now able to force more correct links when there is a strong cohesion indicator; this prevents entities being divided and results in the output showing a greater recall of correct clusters.

Considering the errors made by our system and IMS, it is not clear that one system is clearly better than the other. On gold, our Mutual Information model makes 44 more divided entity errors and 17 more missed entities than IMS, but fewer errors on the remainder of categories. On auto, we make 109 more divided entity errors, 5 more extra

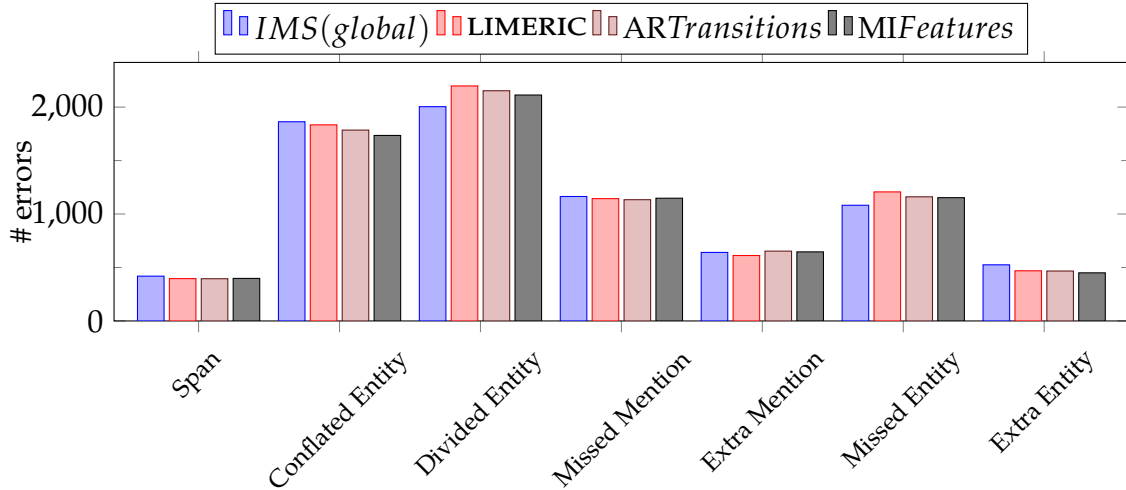


Figure 6.3: Errors made by our Mutual Information model compared to our previous models and IMS on CoNLL-2012 TEST using automatic preprocessing.

mention errors, and 71 more missed entities but outperform IMS on the remaining four error categories. Our conclusion is that while we achieve a higher CoNLL score on gold preprocessing and IMS achieves a higher CoNLL score on automatic preprocessing, neither system can be said to be better. Rather, we merely trust gold information more, and are more conservative on the automatic setting, than IMS. We therefore expect the performance of LIMERIC to improve with advances in upstream processing.

## 6.5 Summary

This chapter introduced a means of studying the mutual information between features in a coreference system using the  $\chi^2$  distribution. We presented an analysis of the mutual information of LIMERIC’s feature set and used this analysis both to motivate a series of secondary features, as well as to highlight areas requiring further study.

The mutual benefit from our secondary competition features and the incorporation of anaphoricity competition into our framework outperformed our AR Transitions model by 0.33% and 0.61% on CoNLL-2012 using gold and automatic preprocessing. Our improvement on the automatic setting was significant at the conservative level of

$p = 0.01$  and resulted in a system which was not significantly different from the strong baseline set by Björkelund and Kuhn (2014).

In terms of further study, we highlighted an important shortcoming of our current feature set: while cohesion appears to be a necessary condition for coreference, it is not sufficient. Difficult cases had contextual cues of coreference, which we use to motivate our exploration of frame semantic inference in the next chapter.

## 7 Frame Semantic Inference

This chapter extends from our analysis in the last, exploring the problem of how to use contextual information, specifically frame semantic knowledge, to improve the resolution of linguistically compatible mentions. While designated by Accessibility theory as a key factor in mediating reference resolution, inference is difficult to encode computationally and we analyse the particular challenges here.

The main contribution of this chapter is our characterisation of frame semantic inference as a two stage process, involving *predicate clustering* and *argument selection*. This characterisation allows us to describe the different challenges for coreference resolution in the general task of OntoNotes and the specialised task of the Winograd Schema Challenge. We find that predicate clustering is particularly a challenge for OntoNotes, since we must account for a full document context and available resources have limited coverage and additionally may not capture narrative structure. However, syntactic parallelism works as a reasonable baseline approach to argument selection given OntoNotes’ natural discourse settings.

We use our analysis to propose Brown clusters as a suitable, and readily available, alternative to traditional frame semantic resources. The gains we see with Brown cluster features open the possibility of exploring frame semantic features for coreference resolution in under-resourced settings.

## 7.1 Background

By focusing on the problem of frame semantic inference, the related task of the Winograd Schema Challenge (WSC; Levesque, 2011) becomes relevant. While our aim remains to improve our understanding of the standard coreference task of OntoNotes, the WSC is an interesting complementary benchmark since it targets challenging resolutions which depend crucially on contextual cues. Systems competing on this task typically augment coreference feature sets by consulting external resources such as FrameNet (Baker et al., 1998) and Narrative Schemas (Chambers and Jurafsky, 2010) and we test and adapt these features in our experiments over OntoNotes. Novel to this work, we additionally consider Brown clusters (Brown et al., 1992) as a potential source of frame semantic knowledge, and motivate this decision below.

### 7.1.1 Winograd Schema Challenge

The Winograd Schema Challenge was introduced by Levesque (2011) as an alternative to the Turing test (Turing, 1950) for assessing whether a computational system has achieved human-level intelligence. The challenge is formulated via a series of sentence pairs with properties demonstrated by the following example.

The trophy would not fit in **the suitcase** because **it** was too small.  
**The trophy** would not fit in the suitcase because **it** was too large.

In this pair, the resolution of the anaphor ‘it’ is ambiguous between the two preceding antecedent choices, one of which is indicated contextually. The correct antecedent in each sentence is indicated in bold and resolution relies on the understanding that containment requires a larger object to enclose a smaller one.

Common Sense Reasoning<sup>1</sup> organises shared tasks testing the WSC, but the primary dataset evaluated in the coreference resolution literature is that of Rahman and Ng (2012). The dataset comprises 943 sentence pairs following the format above composed

---

<sup>1</sup><http://commonsensereasoning.org/winograd.html>

by 20 students in an undergraduate computing class. Each sentence contains an ambiguous pronoun with two candidate antecedents. All systems discussed below, which form our benchmark of current approaches to the WSC, are tested on the same TRAIN and TEST splits of this dataset.

### 7.1.2 Frame Semantic Resources

Frame semantic information was important in early theorising of language understanding (e.g. Schank and Abelson, 1977) since it encodes inference decisions made by humans processing discourses about developing events.

#### FrameNet

FrameNet (Baker et al., 1998) is a collection of manually annotated frames: predications with parallel syntactic-semantic constructs. Predicates (typically verbs) are arranged into equivalence classes called frames wherein each unit templates equivalent events. For instance, *'attack'* and *'bomb'* co-occur in the *'Attack'* frame since both involve a sentient *'Assailant'* injuring a sentient *'Victim'*. *'Assailant'*, *'Victim'*, and other semantic roles are annotated for all frames, though there is no simple mapping from grammatical argument to semantic role.

Consistent with predictions of grammatical parallelism, Rahman and Ng (2011) suggest that FrameNet data may be relevant to coreference since coreferential mentions should fill the same role throughout a discourse. To model role using grammatical arguments, Rahman and Ng defines a sparse feature whose values are triples with the first element being whether or not the governing verbs of two mentions are in the same frame, and the second and third being the grammatical arguments of the two mentions. This feature increases  $B^3$  and CEAFF by 0.4 and 0.3 on the newswire documents common between OntoNotes 2 and ACE 2004/2005.

## Narrative Cloze

The Narrative Cloze task was introduced by Chambers and Jurafsky (2009) to assess how well computational systems can model consistency in narrative structure. In particular, the task measures how well a missing verbal predicate can be predicted by a system with access to its document context. Narrative Cloze is relevant to coreference resolution in that it requires systems to infer that chains of verbal predicates with coreferential arguments are related to one another. That is, while Narrative Cloze uses coreference to find chains of verbal predicates, we would like to use knowledge of related predicates to inform the coreference of mentions.

The dataset used has been that of Chambers and Jurafsky (2010) which was automatically extracted from the New York Times (NYT) portion of Gigaword<sup>2</sup>. The release comprises two types of datasets. For ease of exposition in describing these, we will refer to the pair of a mention's grammatical argument and its governing verb as its *predicate frame*. For instance, the predicate frame for '*the attack*' mention in '*the attack killed 17 American soldiers*' is '*kill-s*' since '*the attack*' is the subject of the predicate '*killed*'.

*Schemas* are defined to be sets of predicate frames related by narrative structure. For instance, the set {'*raise-s*', '*cut-s*', '*increase-s*', '*lower-s*', '*reduce-s*', '*boost-s*'} constitutes a schema since each of these verbs are related by tending to take arguments (in this case, subjects) which are coreferential with one another. schema datasets are given for schema sizes 6, 8, 10, and 12, with larger schemas consuming smaller ones, expanding the set of related predicates, but being based on fewer instances in NYT.

The release also contains a verb-pair dataset which is intended to assist with inferring a natural ordering over predicate frames in schemas. Verb pairs are ordered pairs of verbal predicates (i.e. not predicate frames) with a frequency count, where the count indicates how often a trained classifier predicted the given ordering reflected the true temporal ordering of the predicated events. In this way, if the count of a pair

---

<sup>2</sup><https://catalog.ldc.upenn.edu/LDC2003T05>

$(a, b)$  is much greater than that of  $(b, a)$ , it can be said with reasonable confidence that  $a$  precedes  $b$  temporally.

Irwin et al. (2011) use this data for their CoNLL-2011 system in a feature encoding whether mentions' predicate frames co-occur in a schema<sup>3</sup>. Unfortunately, this feature is not analysed in their ablation study. Rahman and Ng (2012) also use the schema data in their approach to the Winograd Schema Challenge. Like Irwin et al., their feature is based on whether two mentions' predicate frames co-occur in a schema, but inference is extended. Specifically, if a candidate antecedent's partner frame (i.e. with '-s | -o' replaced by '-o | -s') also co-occurs with that of the mention in any schema, no feature is generated.

Although not documented in Rahman and Ng (2012), their reported results for Narrative Schema were also based on lexicalised features as well as one using the verb-pair dataset (personal communication with the authors). Their verb-pair feature is boolean valued to indicate whether the verbs governing two mentions are more likely in the document order or reversed, based on the provided frequency counts. Removing this collection of features in ablation showed a drop of 4.8% accuracy and a decision tree trained over just these features correctly resolved 30.67% of sentences in the Winograd test dataset, incorrectly resolved 24.47%, and was unable to make a prediction on the remaining 44.86%. The system was not tested on OntoNotes.

Peng et al. (2015) explore using the confidence values assigned to schemas as features for modelling competition between candidate antecedents. When two antecedents of a mention are related via a schema, they incorporate the corresponding confidence values into their system in two ways. Firstly, they add the values themselves, weighted by a manually tuned parameter, to the classifier score. Secondly, they use a constraint term in their integer linear programming formulation which says that if the confidence of one schema pairing is higher than another pairing, coreference cannot be resolved between the lower confidence pair without also being resolved for the higher confidence

---

<sup>3</sup>Irwin et al. (2011) and Rahman and Ng (2012) both use `schema-size12`.

pair. These measures improve precision on the Winograd schema Challenge by 23%, but diminish performance on CoNLL by 0.3%.

### 7.1.3 Brown Clustering

Brown clustering (Brown et al., 1992) is an approach to hierarchically clustering words according to either their semantic or syntactic similarity. Each node in the produced (binary) tree is associated with a set of words which are functionally equivalent to one another, with respect to the partitioning. Nodes which are close to each other in this tree, such as siblings, are so placed because their word sets have high pairwise information. For instance, in the Turian et al. (2010) release of Brown clusters, *'said'* and the mis-spellings *'siad'* and *'ssaid'* are in the same cluster, and this cluster has a sibling cluster containing the related terms *'insists'*, *'conceded'*, and *'reasserts'*.

Each word is given an identifier which represents a path of left and right child transitions from the root to the node it has been assigned to. Identifiers are typically given as bitstrings whose 1s and 0s represent the two parent-to-child transitions. In this way, nodes which are close to one another will have similar identifiers and this can be exploited for determining the similarity of words. Given the hierarchical nature of the clustering, neighbourhoods of related nodes are typically defined by taking prefixes of cluster identifiers: a prefix gives a path to non-leaf node and all terms in nodes under this target node will have the same prefix.

There has yet to be a study exploring whether their encoding of distributional semantic means that Brown clusters implicitly encode frame semantic information. However, this seems reasonable given that verbs which tend to take similar arguments should be assigned to nearby clusters. We explore the Brown clusters in Turian et al.'s (2010) release, which includes 1000 and 3200 cluster outputs, as automatically extracted sources of semantic frame data.

## 7.2 Frame Semantic Resources

In this section, we compare the three resources we exploit for frame semantic information with the goal of understanding their similarities and differences for feature development. We are particularly interested in exploring how they differentiate between the Winograd Schema Challenge and OntoNotes datasets, since we would like to understand how to leverage the strengths of WSC features for OntoNotes.

We guide our analysis by considering frame semantic inference to be a two-stage process. In particular, we see all features reviewed above to comprise the stages *predicate clustering* and *argument selection*. For a feature to indicate coreference between two mentions, their governing predicates must first be considered related by a resource, then the mentions must appear as related arguments. That is, deriving frame semantic features can be thought of as a filtering process: relationships between mentions with incompatible governing relationships are ruled out, before those in incompatible grammatical positions. The remainder are suitable candidates for coreference.

### 7.2.1 Predicate Clustering

In predicate clustering, groups of textual predicates are identified as related for inferring coreference relationships. Rahman and Ng (2011) use FrameNet co-occurrence to cluster predicates describing similar events, while Rahman and Ng (2012) and Peng et al. (2015) use schema co-occurrence, thereby clustering predicates related by narrative structure. We propose having the same Brown cluster identifier (or prefix thereof) should likewise cluster predicates as related. We note that relating predicates means that there is likely to be a coreference relationship among certain of their arguments; it does not tell us which arguments are coreferential. Argument selection applies after predicate clustering to make these decisions.

In order for two predicates to be clustered by a resource, both predicates need to be covered by that resource and the pair need to be marked as related. We quantify

Resource	PropBank		Dependencies		Resource	Gold
	Gold	System	Gold	System		
Mentions	19156	47335	19156	47335	Mentions	3762
Predicated	53.6	40.7	55.2	44.2	Predicated	99.1
schema-6	30.8	24.0	31.6	25.2	schema-6	66.7
schema-8	32.1	24.5	32.9	25.7	schema-8	65.8
schema-10	31.9	24.2	32.7	25.4	schema-10	64.5
schema-12	30.9	23.6	31.8	24.8	schema-12	63.3
verb-pair	49.8	37.4	46.5	35.9	verb-pair	77.6
FrameNet	30.7	23.1	32.0	24.8	FrameNet	55.5
Brown	52.8	40.2	55.7	43.5	Brown	93.7

(a) OntoNotes 5
(b) WSC

Table 7.1: Coverage of mentions by the proposed frame semantics resources.

coverage via mentions: a mention is covered if it is the argument of a predicate which is described by a given resource. For FrameNet and Brown clusters, this simply means that the predicate appears in at least one frame or appears in a Brown cluster<sup>4</sup>. For schema it additionally requires the mention to have the correct grammatical argument given the predicate frames described by the resource. To get a coarse-grained idea of coverage for verb-pair, we label a mention as covered if its governing verb is a member of any given verb pair, noting that this may overestimate the rate at which the resource can actually be applied.

We identify two ways of establishing predicate argument structure. First, PropBank-style annotations (Kingsbury and Palmer, 2002) are provided with OntoNotes; while these are not complete, Hovy et al. (2006) expect them to have good coverage of relationships between noun arguments (mentions) and their verbal predicates. However, PropBank annotations are not available for WSC and Rahman and Ng (2012) and Peng et al. (2015) instead use dependency annotation for predicate argument structure over the dataset. We compare coverage from the PropBank annotations against using Stan-

<sup>4</sup>We use the 3200 cluster data of Turian et al. (2010).

ford dependency labels produced automatically with CoreNLP<sup>5</sup> (Chen and Manning, 2014). Mentions are labelled as covered by a resource if their head is linked by an ‘*nsubj*’, ‘*dobj*’, ‘*nsubjpass*’, or ‘*agent*’ dependency arc to a token described in the resource.

Coverage statistics for gold and system mentions are summarised in Table 7.1. The first line gives the raw number of mentions considered and the second gives the proportion of these mentions which are aligned to either PropBank or a dependency arc, as an upper bound of coverage. Considering the OntoNotes data first, we can see that our upper bound is higher when we use dependencies compared to PropBank, though the difference is small and diminished when we consider the coverage of resources themselves. Therefore, we expect PropBank annotations to be sufficient for defining coreference features for OntoNotes, though dependencies may offer slight benefits in coverage. In development, we found PropBank-based features almost consistently outperformed dependency-based ones and analysis suggested this was due to noise from incorrect grammatical structures in the dependency annotation.

Across the resources, coverage ranges from almost complete with Brown clusters to just over half with schema. The low coverage of schema and FrameNet is our first limit on the applicability of frame semantic features for OntoNotes: features cannot be defined for mentions if they are not even considered in predicate clustering.

Looking now at the WSC data, we can see that almost all mentions are linked to a predicate, which is reasonable given that these sentences tend to be short and have simple syntactic structure. With the exception of Brown clusters, which again has almost complete coverage, frame semantic resource cover around two thirds the predicates seen. That is, based purely on coverage, we expect frame semantic resources to be less informative for OntoNotes than WSC.

Next, we quantify how well the relationships between predicates covered in our resources translate to clustering mentions that are coreferential. We expect a resource is good at describing coreference relationships if most coreferential mention pairs

---

<sup>5</sup><http://nlp.stanford.edu/software/corenlp.shtml>

Resource	Coreference		Non-Coreference	
	Pairs	Covered	Pairs	Covered
schema-6	3301	15.7	5349	2.2
schema-8	3505	21.2	5589	3.0
schema-10	3479	21.5	5555	3.5
schema-12	3361	25.1	5377	4.4
verb-pair	6273	48.6	8955	26.5
FrameNet	3386	29.6	5339	6.5
Brown	6744	31.2	9530	7.4

(a) OntoNotes 5

Resource	Coreference		Non-Coreference	
	Pairs	Covered	Pairs	Covered
schema-6	515	7.0	518	6.4
schema-8	497	9.3	506	8.9
schema-10	484	10.5	498	9.6
schema-12	469	11.9	474	10.5
verb-pair	743	38.1	739	38.3
FrameNet	363	17.9	370	17.0
Brown	1113	16.9	1111	16.2

(b) WSC

Table 7.2: Coverage of mention-pair links by the proposed frame semantics resources.

are related by that resource and most non-coreferential pairs are covered but not related. For FrameNet and the Brown cluster data, we label two mentions as related if their governing verbs belong to the same frame or have the same cluster identifier. Relatedness in the Narrative Schema dataset requires mentions' predicate frames or governing verbs to co-occur in a schema or verb-pair.

To assess the margin distinguishing coreference and non-coreference instances, we first collect all (gold) mentions labelled as covered above using PropBank annotations. For entity clusters with two or more mentions covered, we iterate over each covered mention, tallying whenever there is a resource link between the mention and any of the mentions preceding it in the cluster. For each mention, including cluster-initial mentions, we additionally tally whether it and its closest non-coreferential antecedent in the document share a resource link.

Table 7.2 shows us that, similarly for mention coverage, the coverage of links is low across all resources and both datasets. For instance, among the 19156 gold mentions in OntoNotes, respectively, only around 30% (~5700) were covered by a schema. Of these 5700 mentions, only ~3400 coreference pairings exist and only 20% or fewer of these pairings have predicate frames which co-occur in schema. Such low resource

coverage acts as a strict filter on how applicable frame semantic features can be to OntoNotes. However, the filter is reasonably precise. On OntoNotes, all resources have a 20% or greater margin distinguishing coreferential and non-coreferential instances. This suggests that predicate clustering over the context of a whole document is itself informative, and that the presence of a resource link can potentially add information to a coreference model.

This is not the case on WSC, for which we see link coverage to be relatively even between the coreferential and non-coreferential instances. That no margin is seen for this dataset makes sense since the two candidate antecedents are typically sibling arguments of the same predicate. However, it does mean that predicate clustering alone cannot be used for the difficult dataset, necessitating sophisticated reasoning for argument selection.

### 7.2.2 **Argument Selection**

Once mentions are identified as candidates for coreference resolution by predicate clustering, their grammatical role is used to make the final decision about which should be related. Rahman and Ng's (2011) learned to select between arguments of clustered predicates by concatenating the argument numbers of candidate mention pairs. On the other hand, argument selection in Rahman and Ng (2012) leverages the grammatical information encoded in the predicate frames of schema: their schema features only triggered for mentions in grammatical positions precisely identified as related by the resource.

While complex reasoning appears necessary for resolving ambiguous pronouns in the Winograd Schema Challenge, we would expect syntactic parallelism to account for a larger proportion of cases in natural discourse settings, such as those in OntoNotes. That is, without any other information, we expect two subject mentions or two object mentions to have an increased chance of being coreferential with one another based on their grammatical position.

Resource	Arguments				Resource	Arguments			
	ss	oo	so	os		ss	oo	so	os
schema-6	313	204	0	0	schema-6	219	56	182	58
schema-8	500	238	1	5	schema-8	210	57	175	55
schema-10	502	241	2	3	schema-10	204	56	169	55
schema-12	578	255	4	8	schema-12	194	56	165	54
verb-pair	1927	355	361	404	verb-pair	328	71	275	69
FrameNet	712	217	31	41	FrameNet	143	45	128	47
Brown Clusters	1136	809	75	86	Brown Clusters	485	90	447	91

(a) OntoNotes
(b) WSC

Table 7.3: Number of subject and object mentions in pairs related by the proposed frame semantic resources.

To test how well syntactic parallelism works as a simple inference strategy for related predicates, we take the pairs related via PropBank annotations above, and tally the grammatical arguments of the coreferential mentions. The columns of Table 7.3 are labelled by concatenating the labels of the mention (first) and the antecedent (second).

In Table 7.3a, we can see that, as expected, there is a strong tendency for coreferential mentions related by *schema*, *FrameNet*, and *Brown* clusters to share argument number. *schema* and *FrameNet* particularly relate subject mentions, while *Brown* clusters relate mentions of both grammatical positions. We therefore expect simple syntactic parallelism features, enriched with predicate clustering information from these resources, to help coreference resolution on *OntoNotes*.

The tendency for syntactic parallelism is also attested, though less prominently, in pairs from *verb-pair*. With its higher coverage, it could be expected to encompass a wider range of discourse phenomena than *schema* and *FrameNet* and therefore benefit from hand-coded or learned inference rules. That *Brown* clusters appear not to necessitate this inference is interesting, and potentially a byproduct of how the resources were created.

	FrameNet	schema	pair	Brown
FrameNet	-	0.6	11.2	5.4
schema	1.6	-	21.7	16.9
pair	0.1	0.4	-	20.8
Brown	0.0	0.0	6.4	-

Table 7.4: Overlap between proposed frame semantic resources.

On the other hand, Table 7.3b shows syntactic parallelism cannot be assumed for WSC mentions governed by related predicates. This again seems reasonable, given the aim of the dataset to be a collection of particularly difficult instances of coreference, and licenses the sophisticated inference features proposed for the dataset.

### 7.2.3 Inter-Resource Comparison

We have seen that each of our target resources is different in its coverage both of individual predicates in the OntoNotes and WSC datasets, as well as the links required to cluster predicates for inference. We now compare the coverage of our resources directly against one another to shed light on whether they are overlapping or complementary. We also do this with the goal of finding further validation to the statistics above that Brown clusters encode similar information to that in traditional resources.

Table 7.4 measures the overlap between two resources, *A* (vertical) and *B* (horizontal) by iterating over the terms of predicates labelled as related in *B* and counting how frequently they are also labelled as related in *A*. *schema* values are given as ranges of minimum and maximum similarity across the four sizes of schema released. We see that 5.4% of terms related via their Brown cluster identifier are labelled as related in FrameNet. This low number makes sense: Brown clusters cluster a large proportion of the English vocabulary where FrameNet targets a class of predicates.

Our first observation is that the overlap between the resources is low overall. Comparing FrameNet to *schema*, as gold and silver standard datasets, overlap is low in

both directions. We see this complementarity as a good thing: FrameNet describes relationships between similar terms, while schemas find dissimilar terms which tend to be related via a chain of developing events. Their complementarity opens the door for each to model separate aspects of frame semantic inference. Between schema and verb-pair, there is reasonable overlap, presumably from their generation from the same corpus.

We can also see that terms related by Brown clustering are unlikely to be related in FrameNet or Narrative Schema, though these low probabilities may be due the larger vocabulary in Brown clusters. Indeed, terms related by the traditional frame semantic resources *are* likely to be in related Brown clusters, and the proportion of overlap reflects the relative sizes of these resources. That is, the information encoded in both FrameNet and Narrative Schemas is at least partially also encoded in Brown clusters, despite these clusters not being explicitly designed to capture this information.

### 7.2.4 Summary

We have decomposed frame semantic inference into two steps, predicate clustering and argument selection which act as filters for feature generation. In this scheme, we find that WSC represents one extreme where predicate clustering is artificially simplified but argument selection is difficult and requires sophisticated inference strategies. On the other hand, predicate clustering is more difficult on OntoNotes, where full document context is available, but the simple heuristic of syntactic parallelism works better for modelling argument selection.

We used our scheme to study the resources and features proposed for the Winograd Schema Challenge. We found predicate clustering to be constrained by resources having low coverage on OntoNotes. One key aim of feature development is therefore to boost coverage and observe the impact of this on system performance. Our analysis highlighted the promise of Brown clusters for deriving frame semantic features.

## 7.3 Feature Development

In this section, we implement features based on our analysis of frame semantic inference as a two stage process of predicate clustering and argument selection. The implementation details below are for mention-pair features; in our entity-level model, feature generation is based on the current mention and its closest antecedent in the candidate antecedent cluster covered by a resource.

By comparing the performance of our reviewed and novel features, we find that the coverage, as well as the precision, of a resource are important for effective predicate clustering. We also see that it is important not only to encode the similarity of predicates, but also the notion of narrative structure, or which predicates tend to follow one another in a cohesive narrative. For these reasons, our strongest performing features are based on Brown clusters and these features dominate the performance of our combined model. We also confirm the validity of syntactic parallelism for argument selection, via our three feature variants.

### 7.3.1 Feature Variants

Our three feature variants generalise the role of grammatical structure for argument selection with frame semantic features. We denote the variants sparse, collapsed, and dense according to the level of grammatical argument structure captured. Sparse features are defined as described for Rahman and Ng’s (2011) FrameNet features. Specifically, sparse features are defined in three dimensions, whether the predicates are linked in the relevant resource, the argument number of the current mention and that of the closest covered antecedent in the candidate entity cluster: `<resource result> + <arg number>i + <arg number>j`, e.g. `match:True+arg0+arg1`. In contrast, collapsed features are defined in two dimensions `<resource result> + <arg comparison>`, where the argument comparison reflects whether the two mentions have the same

grammatical argument number, e.g. `match:True+args:diff`, and dense features in just one, `<resource result>`, e.g. `match:True`.

Interpreting the three variants, `<resource result>` allows us to model predicate clustering, while the remaining dimensions are used for argument selection. If complex inference is required for argument selection, this should be learned using our sparse feature variants. On the other hand, we would expect collapsed feature variants to be most informative in cases where syntactic parallelism can be assumed. Dense variants are included to reduce sparsity given the large number of features LIMERIC is already learning and the poor coverage of frame semantic resources. Models denoted *all* below are trained on all three variants. We report on single variants for our Brown cluster experiments where performance increases are larger, but not on our FrameNet and Narrative Schema experiments, where all perform similarly to each other and to the all variants models.

### 7.3.2 FrameNet

Table 7.5 summarises the performance of our three features based on FrameNet, described below. We can see that each feature improves performance marginally, with our novel frame concatenation and schema clustering features yielding stronger gains than Rahman and Ng’s (2011) same frame when using automatic preprocessing.

The impact of FrameNet features is to improve the link-based MUC and  $B^3$  scores at the expense of the entity-based CEAFE score, netting the modest improvements on CoNLL we see. These changes reflect recall increasing on MUC and  $B^3$  and decreasing on CEAFE, with precision staying relatively constant on all three metrics. Given the increase in link-based recall, we infer that FrameNet features are good at informing the links that they do cover, but this is at the expense of missing entity clusters overall, resulting in the decrease in CEAFE recall.

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
AR + MI	74.08	62.25	60.89	65.74	70.06	57.92	56.36	61.45
Same Frame (all)	74.42	62.51	60.71	65.88	70.19	57.85	56.20	61.41
Concat Frame (all)	74.34	62.62	60.69	65.88	70.38	58.06	56.31	61.58
schema-6 Clustering (all)	74.42	62.63	60.69	65.91	70.29	57.99	56.30	61.53

Table 7.5: Performance of FrameNet features on CoNLL-2012 DEV with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information).

**Same Frame** Re-implementation of Rahman and Ng (2011), whose description describes the sparse variant of this feature. As such, predicate clustering reflects whether the verbs governing two mentions are in the same FrameNet frame. This feature improves system performance by just 0.14%, but only when gold preprocessing is used. Moving to automatic, performance is just weaker than baseline.

Our B<sup>3</sup> increase of 0.26% on gold preprocessing is comparable with the 0.4 and 0.3% improvements reported in Rahman and Ng despite our evaluation being over a stronger baseline. As would be expected given our feature set, this suggests our baseline improvements are not capturing the frame semantic regularities of FrameNet.

**Frame Concatenation** Where same frame captures parallelism in grammatical frame, we actually expect that documents in OntoNotes will often describe the progression of a narrative in which multiple events are predicated. We use the concatenation of frame identifiers as the <resource result> of this feature, in order to allow frame compatibilities to be learned. This feature subsumes a sparse version of same frame in that a concatenation is allowed to be between two identical identifiers.

Frame concatenation outperforms same frame by 0.17% using automatic preprocessing. This is encouraging given the sparsity we expect in this feature: with 1020 distinct frames, there are  $1020^2 = >1\text{M}$  possible frame pairs. This promising result shows

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
AR + MI	74.08	62.25	60.89	65.74	70.06	57.92	56.36	61.45
Same schema-10 (all)	74.51	62.56	60.61	65.89	70.24	57.69	56.38	61.44
verb-pair Order (all)	74.29	62.43	60.69	65.80	70.20	57.84	56.22	61.42
Same verb-pair (all)	74.52	62.50	60.96	65.99	70.35	57.87	56.31	61.51

Table 7.6: Performance of Narrative Schema features on CoNLL-2012 DEV with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information).

the importance of capturing narrative structure in defining effective frame semantic features.

### Schema Clustering

A less sparse way to encode frame transitions, and to leverage the complementary coverage of FrameNet and Narrative Schemas, is to use schema relatedness to further cluster FrameNet predicates. All predicates in schemas<sup>6</sup> are mapped to frames (where possible) and frame clusters are grown incrementally according to the rule that {A, B, C} is included as a cluster iff all of {A, B}, {B, C}, and {A, C} are. Clusters of the form {A, A} are also included. Co-occurrence in a cluster is used for <resource result> here. We find that schema clustering offers similar improvement above same frame as frame concatenation did, despite its denser representation.

### 7.3.3 Narrative Schema

The performance of our three features based on the Chambers and Jurafsky (2010) datasets are given in Table 7.6. Comparing against our FrameNet results, the gains we see here are similarly modest. However, performance on automatic preprocessing is particularly weak and only our novel feature same verb-pair outperforms baseline.

<sup>6</sup>Experiments here use schema6.

Same verb-pair affects increases on all three metrics on the gold setting, but particularly MUC and B<sup>3</sup>. That is, as for FrameNet features, the effect of using the verb-pair data is to better inform our model at the level of individual links. For CEAFE, recall again falls but precision now increases by 0.26%: using verb-pair features does not compromise our ability to delineate clusters as FrameNet features did.

#### **Same schema**

Based on Irwin et al. (2011) and Rahman and Ng (2012), we use the co-occurrence of two predicates in a schema as the `<resource result>` in this feature. In development, we found schema-10 gave the strongest results, where previous studies use schema-12. We expect the choice between schemas to depend on the relative importance resource coverage and precision in a given setting. That schema-10 outperforms schema-12 here shows that our system is sensitive to non-coreferential instances being falsely indicated.

While not tabulated, the dense formulation of the feature is strong, outperforming the all variants model on automatic preprocessing (CoNLL = 61.54). This is implicitly consistent with Irwin et al. (2011) who devise the feature in this same way. That our dimensions for the grammatical arguments of mentions are not required is presumably due to the fact that matching predicate frames already requires mentions to be in compatible grammatical positions. Adding these extra dimensions merely allows us to learn whether subject or objects are more likely to be related by schema matches.

#### **verb-pair Order**

Re-implementation of the feature used in Rahman and Ng (2012), based on personal communication with the authors. For this feature, `<resource result>` reflects whether the textual order of the two mentions' predicates is consistent with the frequency counts in verb-pair.

While this feature is crucial to the success of narrative schema features in Rahman and Ng (2012), it is actually our weakest surveyed feature. It is possible that this is a coverage effect: the check whether textual order is consistent with the given frequency counts reduces the number of mention-pairs which evaluate positively, which means the effective coverage of the resource will be lower than what is estimated in Table 7.2. Given that this coverage was already quite high, we do not expect this to be a full explanation. We suggest that the coherence of the OntoNotes documents also acts as a constraint, naturally guiding the verb pairs we see, making the order check unnecessarily limiting. Additionally, the frequency counts given in `verb-pair` reflect whether the pair is consistent with temporal ordering in the real world, rather than document ordering, which is subject to discourse and pragmatic preferences. To test whether ordering constraints are actually necessary, we formulate our next feature, `same verb-pair`.

#### **Same verb-pair**

We use as `<resource result>` whether two mentions' governing verbs appear at all as a `verb-pair`. We do not check their relative frequency, only imposing a threshold frequency of at least 50 occurrences, where we found best performance. While simple, we find the performance of this implementation is strong, particularly using gold preprocessing. The strength of `same verb-pair` above `same schema` and `same frame` is consistent with our above analysis: `verb-pair` has both better coverage than both other resources. Furthermore, the resource explicitly aims to capture consistencies in the development of narratives, rather than the inherent similarity between predicates.

### **7.3.4 Brown Clusters**

Brown clusters have yet to be explored as a source of frame semantic information for coreference resolution, though we have demonstrated their promise above. As a more direct test, we formulate a feature whose `<resource result>` indicates whether the

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
AR + MI	74.08	62.25	60.89	65.74	70.06	57.92	56.36	61.45
Same 4-Prefix	74.42	62.61	60.61	65.88	70.30	58.10	56.30	61.57
Same 8-Prefix	74.51	62.77	61.05	66.11	70.49	58.16	56.62	61.76
Same 12-Prefix	74.53	62.83	60.87	66.08	70.51	58.35	56.59	61.82

Table 7.7: Performance of Brown cluster features (using sparse representation over 3200 cluster data) on CoNLL-2012 DEV with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information).

governing verbs of mentions has the same Brown cluster identifier, or prefix thereof. We identify three parameters which require exploration: prefix length, degree of clustering, feature variant, and feature variant. To analyse the impact of each on our coreference model, we tune each parameter in turn while holding the other two constant.

### Prefix Length

Table 7.7 shows the performance of sparse features over Turian et al.’s (2010) 3200 cluster dataset; feature values are true if the governing verbs of two mentions have the same Brown cluster identifier up to the prefix lengths given and false otherwise. Using a prefix length of 12 corresponds to the full Brown cluster identifier being used.

Our first observation is that already our Brown cluster features perform better than those we have seen so far for FrameNet and Narrative Schemas. Using the whole cluster identifier, we are 0.20 and 0.24% stronger than frame concatenation and 0.07 and 0.31% stronger than same verb-pair with gold and automatic preprocessing. That Brown cluster features are so strong compared to schema and FrameNet features confirms that a high coverage of coreferential instances is required to model frame semantics. That they outperform features based on the similarly high-coverage verb-pair resource confirms that minimising the number of falsely indicated non-coreferential instances is

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
AR + MI	74.08	62.25	60.89	65.74	70.06	57.92	56.36	61.45
Same 12-Prefix (3200)	74.53	62.83	60.87	66.08	70.51	58.35	56.59	61.82
Same 12-Prefix (1000)	74.67	62.80	61.06	66.18	70.47	58.09	56.59	61.72

Table 7.8: Performance of Brown cluster features (using sparse representation over length 12 prefixes) on CoNLL-2012 DEV with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information).

also important. That is, both the recall and precision of a resource need to be taken into account when selecting frame semantic resources.

We note that this result is also consistent with our suggestion that encoding narrative structure is important for frame semantic features: our analysis indicated that Brown clusters capture aspects of FrameNet and Narrative Schema simultaneously, making them a compact representation of both predicate similarity and narrative structure.

On automatic preprocessing, we see a uniform increase in system performance with increasing prefix length. Since longer prefixes capture more information about word usage, we would expect these features to have higher confidence about relatedness and to minimise noise. Indeed, while precision and recall both increase on all experiments in Table 7.8, the balance shifts from recall to precision as prefix length increases. For instance, the precision increases are 0.17, 0.06, and 0.24% greater than the recall increases on MUC, B<sup>3</sup>, and CEAFE moving to 12-prefixes from 8-prefixes.

Prefix lengths of 8 and 12 both perform well, with the shorter, higher recall prefix giving better results using gold preprocessing and the longer, higher precision prefix giving better results using automatic preprocessing. That is, neither is clearly better but instead make different trade offs between precision and recall.

### Degree of Clustering

We next explore the degree of clustering on performance by comparing features implemented over the 1000 and 3200 cluster datasets of Turian et al. (2010). We use prefix length 12 since this means we are using full identifiers in both instances and the sparse feature variant. As would be expected, the results for the 1000 cluster data in Table 7.8 are similar to what we would expect for 10-prefixes given the results in Table 7.7.

Table 7.8 shows us that, again, neither choice is clearly better. Using a higher degree of clustering (smaller number of clusters) is better when using gold preprocessing, where shorter, higher recall prefixes performed well. Similarly, using a lower degree of clustering (larger number of clusters) is better for automatic preprocessing, where longer, higher precision prefixes performed well. That is, if you have trustworthy preprocessing, a higher degree of clustering, and therefore higher coverage of coreferential instances, can be helpful. Interestingly, the difference in between the two choices in the automatic setting is largely restricted to the  $B^3$  metric, which gives harsher penalties to larger clusters: the drop in performance when moving to the high recall setting derives from errors on topical entities.

### Feature Variant

We consider the impact of feature variant in Table 7.9 using length 12 prefixes over the 3200 cluster dataset. Consistent with our observation that coreferential mentions related by Brown cluster identifiers tended to either be both grammatical subjects or both objects, we find that collapsed is the strongest variant, even outperforming our all variants model. This provides further evidence for the applicability of syntactic parallelism to argument selection in OntoNotes.

The improvement in using the collapsed variant compared to sparse derives from a slight boost on MUC and a larger boost on CEAFE, particularly on precision. Given that CEAFE aims to capture whether a system has returned the right number of entities, we

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
AR + MI	74.08	62.25	60.89	65.74	70.06	57.92	56.36	61.45
Same 12-Prefix (all)	74.55	62.54	60.79	65.96	70.66	58.34	56.65	61.88
Same 12-Prefix (sparse)	74.53	62.83	60.87	66.08	70.51	58.35	56.59	61.82
Same 12-Prefix (collapsed)	74.63	62.82	61.03	66.16	70.57	58.35	56.83	61.92
Same 12-Prefix (dense)	74.55	62.79	60.83	66.06	70.41	58.12	56.39	61.64

Table 7.9: Performance of Brown cluster features (using length 12 prefixes over 3200 cluster data) on CoNLL-2012 DEV with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information).

infer that the mention-pair links corrected by Brown clusters prevent LIMERIC from proposing spurious clusters in this high precision setting.

### 7.3.5 All Resources

The best performing of our features are schema clustering, same verb-pair and same Brown cluster identifier. While frame concatenation performs similarly to schema clustering, the features were designed to encode similar information and we prefer the compactness of schema clustering. We test whether the information encoded in these features above is complementary by first learning a model over the three, and then performing an ablation study. Table 7.10 summarises the results of these experiments.

All resources is the weakest of all the combined models. This is perhaps not unexpected given the very large size of the feature set we now have and our analysis of that the resources overlap, meaning that features defined over them are not independent. Removing schema clustering, we can see that performance is not very different to using just Brown cluster features, though is stronger than our same verb-pair experiments by 0.06% and 0.28%. That Brown cluster features dominate performance here is again not surprising given how well it performs when introduced alone.

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
AR + MI	74.08	62.25	60.89	65.74	70.06	57.92	56.36	61.45
All Resources	74.50	62.63	60.65	65.93	70.53	58.29	56.55	61.79
verb-pair + Brown	74.57	62.60	60.97	66.05	70.56	58.21	56.59	61.79
verb-pair	74.52	62.50	60.96	65.99	70.35	57.87	56.31	61.51

Table 7.10: Performance of combined model using all frame semantic resource features on CoNLL-2012 DEV with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information).

However, our dual-resource model does not quite achieve the performance we saw in Table 7.9. While the small difference may be due to chance effects in training, we manually examine the learned feature weights to investigate any additional factors. We see that between the verb-pair + Brown model and their corresponding single resource models, features achieve remarkably similar weights. That is, in the dual model, we are learning to assign twice the relative importance to frame semantic information.

In cases when only one resource covers a mention pair, it is straightforward that this is the correct solution. For cases in which both resources indicate a match, the model will sum the evidence from both, which results in the match being trusted more. While higher trust is reasonable given that the match is found in two resources, the weight sum might be too highly weighted. Given that only 33% of covered coreferential pairs are related both by verb-pair and Brown clusters, the learned weights are likely to be more correctly tuned for single match than double.

We experiment with combining the models using a feature whose `<resource result>` is true if a match is indicated by either verb-pair *or* same 12-prefix, but find that performance is lower again, 65.97% and 61.79% using gold and automatic preprocessing. We expect the correct solution will involve modelling the non-independence of these features, potentially via secondary feature conjunctions. For these reasons, as well as

the Brown cluster model still being the strongest of all we have seen in development, we benchmark system performance using just our novel Brown cluster features.

## 7.4 Evaluation

This section benchmarks the performance of our novel frame semantic features using Brown clusters. While Brown clusters are derived from unlabelled text, we can not fairly compare against systems developed for the closed version of the shared task since this definition does not allow reference to any external knowledge sources.

### 7.4.1 Benchmarking

We saw above in development of Brown cluster features that there is a recall-precision trade off when using different prefix lengths and, similarly, different degrees of clustering. We therefore submit two systems to testing, a high recall version based on length 8 prefixes and a high precision version based on length 12 prefixes. Table 7.11 shows that, unlike in development, our 8-prefix model outperforms our 12-prefix model on both gold and automatic settings.

Our performance on the different evaluation metrics reveals why our high precision 12-prefix model is not the strongest of our two models. Using gold preprocessing,

	Gold				Auto			
	MUC	B <sup>3</sup>	CEAFE	CoNLL	MUC	B <sup>3</sup>	CEAFE	CoNLL
LIMERIC	73.66	60.64	57.77	64.02	69.74	55.76	53.34	59.61
+ AR Transitions	74.34	<b>61.81</b>	<b>58.74</b>	<b>64.96</b>	70.33	<b>56.71</b>	<b>54.52</b>	<b>60.52</b>
+ Mutual Information	74.73	61.72	59.43	65.29	70.72	57.40	55.26	<b>61.13</b>
+ Frame Semantics (12-prefix)	74.75	62.00	59.75	65.50	70.92	57.38	55.39	61.23
+ Frame Semantics (8-prefix)	74.95	62.16	59.89	65.67	70.92	57.45	55.45	61.27

Table 7.11: Performance of Brown cluster features on CoNLL-2012 TEST with respect to our strongest system from Chapter 6 (AR Transitions + Mutual Information).

we achieve F score gains of 0.28% and 0.32% on  $B^3$  and CEAFF, but MUC does not change appreciably. This is because, while MUC precision increases, as expected, recall decreases. That is, our high precision Brown cluster features make our system even more conservative, adding to this major source of error. With length 12 prefixes on automatic preprocessing, we see a MUC performance gain since precision and recall both increase. We expect the recall gain here is impacted by our previous models for this difficult setting already suffering missing links.

Our 8-prefix model achieves performance gains of 0.38% and 0.14% on CoNLL: the performance gains we saw in development carry over with almost identical magnitude for gold preprocessing. Indeed, we achieve a weakly significant improvement on CoNLL, with  $p = 0.023$ . The gains are across all three metrics, on recall and precision.

### 7.4.2 Error Analysis

Compared to our initial baseline set in Chapter 4 (LIMERIC), we have now achieved performance gains of 1.65% and 1.66% using gold and automatic preprocessing. In previous chapters, we saw that our improvements from incorporating cognitive insights largely derived from reducing the number of conflated and divided entity errors.

Figure 7.1 shows the number of errors made by our same 8-prefix model on CoNLL-2012 TEST using gold preprocessing. We can see that frame semantic features continue to reduce the number of times we conflate entities, as well as miss entities, but other error categories largely are not impacted. The number of divided entity errors we make is only marginally decreased and remains a key source of error.

The error profile in Figure 7.1 also suggests why we found  $B^3$  to be particularly sensitive to modelling changes above (cf. Tables 7.7 and 7.8). Kummerfeld and Klein (2013) demonstrate that the number of conflated errors impacts  $B^3$  precision score more than errors in any other of their categories, but have no impact on  $B^3$  recall. Therefore, we suggest the increases we see in  $B^3$  when optimising for precision above arise from the reduction we now see in conflated entity errors.

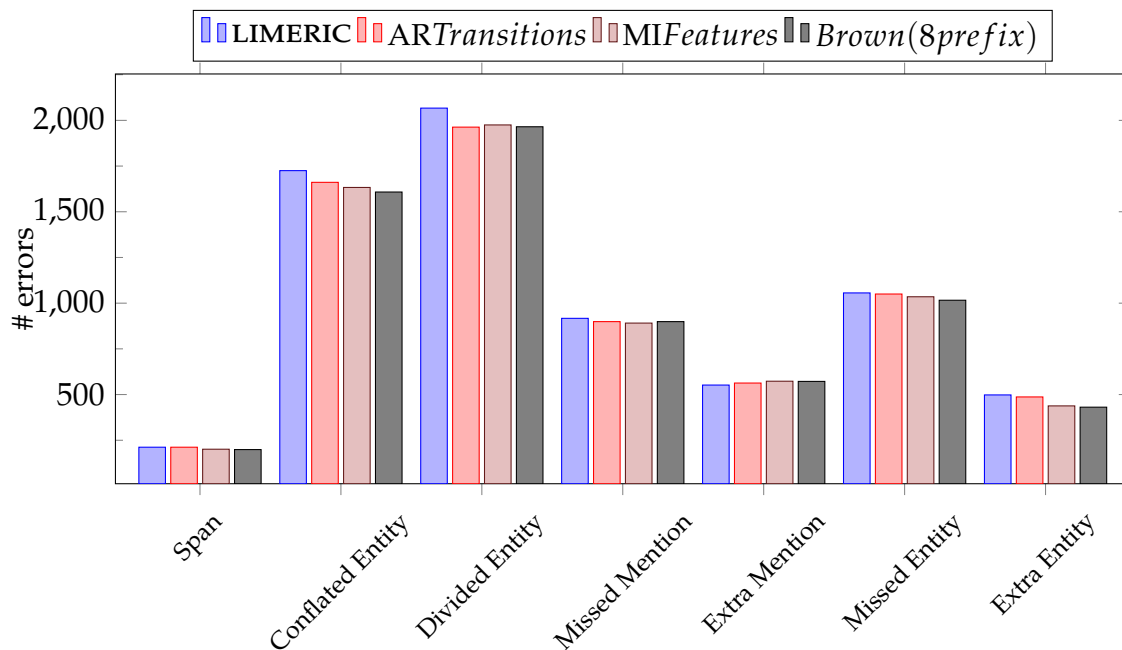


Figure 7.1: Errors made by Same 8-Prefix model compared to our previous baselines on CoNLL-2012 TEST using gold preprocessing.

On the automatic setting in Figure 7.2, we again see that frame semantic modelling has reduced the number of conflated entity errors we make compared to our previously best model. However, this time, we also see slightly more missed mention, extra mention, and missed entity errors. That is, frame semantic features allow us to improve our delineation of entity clusters on both gold and automatic settings, but doing so with automatic preprocessing simultaneously degrades other aspects of our link-based decision making. This accounts for the lower performance gain we see in this setting, and we highlight addressing this problem for future work.

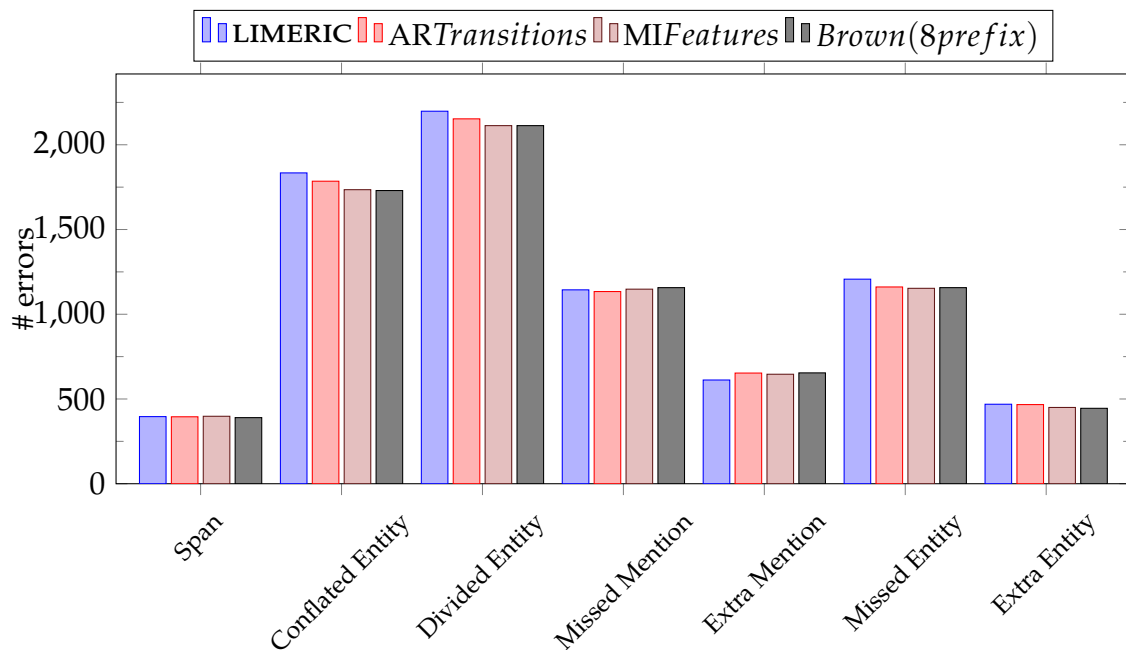


Figure 7.2: Errors made by Same 8-Prefix model compared to our previous baselines on CoNLL-2012 TEST using automatic preprocessing.

## 7.5 Summary

The importance of frame semantic inference to coreference resolution is demonstrated by the specialised task of the Winograd Schema Challenge (WSC). In this chapter, we analyse the relevant differences between the WSC and general-domain coreference in OntoNotes. In doing this, we decompose frame semantic inference into predicate clustering and argument selection in order to understand the current challenges. We find, for features on predicate clustering in OntoNotes to be effective, they need to be based on a resource which encodes narrative structure and has good coverage. Good coverage, in turn, requires high coverage of coreferential instances and low coverage of non-coreferential instances. We additionally show that, in the natural discourse settings in OntoNotes, syntactic parallelism appears to be a satisfactory approximation to argument selection.

These findings are supported by our experimental results. We develop methods for modelling frame semantic inference, adapting previously documented features and proposing novel variants. For the reasons highlighted by our analysis, our novel use of Brown clusters to mine frame semantic information perform particularly well, achieving 65.67% and 61.27% on CoNLL-2012. While marginal, these gains in performance, particularly on the gold setting, are encouraging given that Brown clusters are straightforward to derive from unlabelled text, and open the possibility to explore frame semantic modelling in under-resourced settings.

## 8 Conclusion

Coreference resolution is a complex capability that is an active area of research in both the cognitive and computational literatures. Centering (Grosz et al., 1995) and Accessibility (Ariel, 2001) theories offer cognitive models for understanding how humans resolve reference in natural language, proposing a hierarchy of referring expression types and highlighting the role of cohesion, proximity, parallelism, topicality, competition, and inference automaticity in resolution. Existing computational models leverage some of this insight: systems are typically built around rich sets of cohesion features, but a more limited range of features aimed at the remaining factors.

In this thesis, we design and implement **LIMERIC**, a state-of-the-art coreference resolution engine. Despite its simple model, a baseline feature set achieves the highly competitive performance values of 64.21% and 59.99% using gold and automatic preprocessing on the CoNLL-2012 benchmark task. As well as strong performance, a key contribution of this work is a reconceptualisation of the coreference task. We draw an analogy between shift-reduce parsing and coreference resolution to develop an algorithm which naturally mimics cognitive models of human discourse processing. Leveraging the self-ordering forest of discourse entities as a simple model of the human mind, we propose and validate stack depth as a cognitively aware measure of proximity and use its order directly in our features modelling competition in antecedent selection.

Extending from this strong baseline, we enrich our model using a range of insights offered by cognitive theories. Each contribution yields statistically significant improvements and sum to gains of 1.65% and 1.66% on the CoNLL-2012 benchmark using gold

and automatic preprocessing. Our analysis shows that our final system is either better or not significantly different from the strong baseline set by Björkelund and Farkas (2012). That is, LIMERIC is at once a platform for exploring cognitive insights into coreference and a viable alternative to current systems.

Each novel feature proposed is based on a thorough analysis of its applicability to English OntoNotes. In this way, this thesis contributes to our understanding of the mechanisms underlying reference resolution in real language data. We find fine-grained usage trends that are not expressible in currently used coarse-grained mention typologies, and show that cognitive insights beyond cohesion are required to fully model coreference. We additionally analyse the challenges in applying frame semantic knowledge to coreference resolution.

## 8.1 Future Work

We develop LIMERIC to be highly adaptable for further improvements and anticipate future work extending the ways in which cognitive insights are understood and implemented in computational systems processing real language data.

### 8.1.1 Robust Models of Coreference

In Chapter 4, we presented the simple and intuitive algorithms LIMERIC uses for inference and training. We noted an imperfect but necessary decision to follow gold transitions in training but system predictions at run time. This was because following automatic predictions causes the system to wander off-course quickly and continually re-seeding beams from gold becomes not very different from purely following gold transitions. As feature sets for coreference resolution continue to improve, we expect to see beam search as a means of learning robust models. We also note the cognitive interpretation of such models: they can potentially model cases of ‘near’ coreference, in which resolution may be delayed or corrected as new information is revealed.

At least three other modelling decisions have the potential to be enriched beyond their description here. Firstly, we left shift-reduce parsing with look-ahead to future work; the promising results we show using the simple  $LR(1)$  algorithm suggest that this will be a fruitful direction. Analysis in Chapter 4 demonstrated that the ordering of our forest of discourse entities modelled their salience and that entity-level modelling encodes global consistency constraints. However, we noted that only using proximity to order entities is an approximation of the true ordering. An interesting approach to defining order could be to pose it as a learning to rank problem and using features on entity topicality and other insights from cognitive theories.

In entity-level modelling, we would like to see the number of attributes expressed at the cluster level grow as new features are proposed. For instance, to improve lexical cohesion features, word senses could be defined at the entity-level, also allowing disambiguation to be informed by the types of named entities and pronouns in the same entity cluster as a nominal mention with ambiguous head word.

### 8.1.2 Insufficiency of Cohesion

A key argument of this thesis is that, despite their prominence in the computational literature, cohesion features are not sufficient for modelling coreference. In Chapter 4, we see that the performance of cohesion features plateau as they broaden to capture fuzzier relationships, while our feature association analysis in Chapter 6 demonstrates that both surface form and linguistic attribute cohesion cannot be fully understood without taking into account their interaction with a range of conflating factors including proximity and topicality.

Accounting for the factors that conflate and nuance head match, a commonly-used and powerful indicator of coreference, was particularly challenging. We identified a variety of factors including mention referentiality, restrictive modification, as well as the head word itself and the genre of its document. While we designed second-order features to target these subtleties, a more complete solution could be to build separate

classifiers for a number of these decisions. We especially see the classification of a mention as referential, vague, or generic as promising, particularly given that this is related to the problem of singleton detection in OntoNotes.

An alternative research direction which arises from the analysis in Chapter 6 is to explore how basing features on automatic preprocessing affects the associations observed since, for instance, NER is core to how linguistic attributes are determined. Such analysis would potentially shed light on how to design systems to be robust to upstream annotation errors, which LIMERIC appears to be more sensitive to than the approaches of Fernandes et al. (2012) and Björkelund and Kuhn (2014).

### 8.1.3 Extending Frame Semantic Inference

We used our analysis in Chapter 6 to motivate our study of frame semantic inference in Chapter 7. We had some success in these experiments, finding that the performance of frame semantic features reflected the coverage of their resources, as well as their level of noise and whether they encoded narrative structure. For these reasons, features based on Brown clusters performed well. Given the ease in extracting Brown clusters from unlabelled text, this promising result opens the possibility for exploring frame semantic features in under-resourced settings.

However, there is certainly room for improvement. Foremost, despite their coverage only partly overlapping, we were unable to find mutual benefit from using multiple resources. We therefore leave as future work how best to combine different frame semantic information so that they produce complementary benefits in sum. Additionally, our most impactful features on TEST were high recall variants, arguing for the use of higher coverage resources. While investing in larger hand-curated frame resources will undoubtedly enhance our understanding of the inference we wish to capture, the good performance of our Brown cluster features suggests that we should also explore automatic, distributional methods for mining such information.

### 8.1.4 Further Insights from Psycholinguistic Theories

We showed in Chapter 4 that a cognitive measure of proximity, depth, was both sound and able to be learned by our system. Further to this, both Centering and Accessibility theories emphasise the importance of discourse segmentation on perceived proximity. We expect richer models of proximity to be especially relevant for newswire documents comprised of squibs, and longer texts such as reports, essays, and novels.

Similarly, topicality is only explored insofar as mentions which are grammatical subjects or members of large entity clusters are topical. Since entity clusters grow incrementally, this means information on the topicality of entities is unreliable at the beginning of documents. This underspecification could be complemented by document-level or collection-level topicality measurements.

Additionally, frame semantics is only one of many cues on which humans base their inference of reference expression referent. Others include world knowledge, discourse relations, and pragmatic goals. All of these are promising directions for future research.

### 8.1.5 Languages Other Than English

Given that much of this thesis has been motivated by cognitive models of discourse processing, we would expect it to be a useful foundation for studying coreference in languages other than English. We have already identified that using Brown clusters to mine frame semantic information is particularly powerful since they can be generated for under-resourced settings, which includes under-resourced languages. We also see the methodology we introduce in Chapter 6 for understanding feature interactions to be completely language-independent: a coreference resolver trained for any language can be studied using our techniques. Comparing association statistics across resolvers for different languages would allow us to identify any universals in how languages indicate coreference.

The shift-reduce inspired algorithm we propose in Chapter 4 should similarly require little update to apply to non-English corpora. This expectation is based on the assumption that discourse develops in all languages by referring to previously mentioned entities and concepts. Factors which could vary between models for different languages could be the rate of singleton mentions and the expected depth in the discourse stack of antecedents, both of which are learned during training (rather than being manually specified).

Our work extending the Accessibility hierarchy from spoken Hebrew to written English demonstrates that the approach is valid across languages, even those belonging to different language families. While different mention classification schemes may be required, our methodology for exploring systematic patterns is language-independent once a classification scheme has been implemented.

## 8.2 Summary

Coreference resolution remains an active area of research and our work has provided a simple alternative for approaching the task. We have addressed a number of shortcomings in applying cognitive insights to computational models of coreference resolution, but a number of challenges remain. We are excited by the promise of Accessibility theory in formalising the challenges which remain. By expanding our understanding of how best to model coreference, we improve our ability to understand the meaning of natural language texts and to organise and leverage the information they express.

# Bibliography

The 6th Message Understanding Conference. 1995. *Coreference Task Definition*.  
[http://www.cs.nyu.edu/cs/faculty/grishman/COTask21.book\\_1.html](http://www.cs.nyu.edu/cs/faculty/grishman/COTask21.book_1.html).

The 7th Message Understanding Conference. 1997. *MUC-7 Coreference Task Definition*.  
[http://www-nlpir.nist.gov/related\\_projects/muc/proceedings/co\\_task.html](http://www-nlpir.nist.gov/related_projects/muc/proceedings/co_task.html).

Alfred V Aho and Stephen C Johnson. 1974. LR parsing. *Association for Computing Machinery Computing Surveys*, 6(2):99–124.

Gabor Angeli, Arun Chaganty, Angel Chang, Kevin Reschke, Julie Tibshirani, Jean Wu, Osbert Bastani, Keith Siilats, and Christopher D Manning. 2013. Stanford’s 2013 KBP System. In *Proceedings of the 2013 Text Analysis Conference*.

Mira Ariel. 2001. Accessibility theory: An overview. *Text Representation: Linguistic and Psycholinguistic Aspects*, pages 29–87.

Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *Proceedings of the 1st International Conference on Language Resources and Evaluation: Workshop on Linguistic Coreference*, pages 563–566.

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The Berkeley FrameNet project. In *Proceedings of the 17th International Conference on Computational Linguistics*, pages 86–90.

Breck Baldwin. 1997. CogNIAC: high precision coreference with limited knowledge and linguistic resources. In *Proceedings of a Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*, pages 38–45.

Breck Baldwin, Tom Morton, Amit Bagga, Jason Baldridge, Raman Chandraseker, Alexis Dimitriadis, Kieran Snyder, and Magdalena Wolska. 1998. Description of the UPenn CAMP system as used for coreference. In *Proceedings of the 7th Message Understanding Conference*.

BBN Technologies. 2012. *OntoNotes Release 5.0*.

Eric T Bell. 1934. Exponential numbers. *American Mathematical Monthly*, pages 411–419.

Eric Bengtson and Dan Roth. 2008. Understanding the value of features for coreference resolution. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 294–303.

Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. 2012. An empirical investigation of statistical significance in NLP. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 995–1005.

Anders Björkelund and Richárd Farkas. 2012. Data-driven multilingual coreference resolution using resolver stacking. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and the Conference on Natural Language Learning: Shared Task*, pages 49–55.

Anders Björkelund and Jonas Kuhn. 2014. Learning structured perceptrons for coreference resolution with latent antecedents and non-local features. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 47–57.

- Peter F Brown, Peter V Desouza, Robert L Mercer, Vincent J Della Pietra, and Jenifer C Lai. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.
- Jie Cai and Michael Strube. 2010. Evaluation metrics for end-to-end coreference resolution systems. In *Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 28–36.
- Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised learning of narrative schemas and their participants. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the Association of Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation for Natural Language Processing*, pages 602–610.
- Nathanael Chambers and Dan Jurafsky. 2010. A database of narrative schemas. In *Proceedings of the 7th International Conference on Language Resources and Evaluation*, pages 1614–1618.
- Kai-Wei Chang, Rajhans Samdani, and Dan Roth. 2013. A constrained latent variable model for coreference resolution. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 601–612.
- Kai-Wei Chang, Rajhans Samdani, Alla Rozovskaya, Nick Rizzolo, Mark Sammons, and Dan Roth. 2011. Inference protocols for coreference resolution. In *Proceedings of the 15th Conference on Computational Natural Language Learning: Shared Task*, pages 40–44.
- Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 173–180.

- Danqi Chen and Christopher D Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 740–750.
- Yoeng-Jin Chu and Tseng-Hong Liu. 1965. On the shortest arborescence of a directed graph. *Science Sinica*, pages 1396–1400.
- Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pages 1–8.
- Koby Crammer and Yoram Singer. 2003. Ultraconservative online algorithms for multiclass problems. *The Journal of Machine Learning Research*, 3:951–991.
- Aron Culotta, Michael Wick, Robert Hall, and Andrew McCallum. 2006. First-order probabilistic models for coreference resolution. In *Proceedings of the 2007 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 81–88.
- Hal Daumé III and Daniel Marcu. 2005a. A large-scale exploration of effective global features for a joint entity detection and tracking model. In *Proceedings of the 2005 Joint Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 97–104.
- Hal Daumé III and Daniel Marcu. 2005b. Learning as search optimization: Approximate large margin methods for structured prediction. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 169–176.
- Pascal Denis and Jason Baldridge. 2007. A ranking approach to pronoun resolution. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 1588–1593.

- Pascal Denis and Jason Baldridge. 2008. Specialized models and ranking for coreference resolution. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 660–669.
- George R Doddington, Alexis Mitchell, Mark A Przybocki, Lance A Ramshaw, Stephanie Strassel, and Ralph M Weischedel. 2004. The automatic content extraction (ACE) program-tasks, data, and evaluation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation*, pages 837–840.
- Greg Durrett and Dan Klein. 2013. Easy victories and uphill battles in coreference resolution. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1971–1982.
- Jack Edmonds. 1967. Optimum branchings. *Journal of Research of the National Bureau of Standards*, pages 233–240.
- Gilles Fauconnier. 1994. *Mental spaces: Aspects of meaning construction in natural language*. Cambridge University Press.
- Gilles Fauconnier and Mark Turner. 2008. *The way we think: Conceptual blending and the mind's hidden complexities*. Basic Books.
- Eraldo Fernandes, Cícero dos Santos, and Ruy Milidiú. 2012. Latent structure perception with feature induction for unrestricted coreference resolution. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Conference on Natural Language Learning: Shared Task*, pages 41–48.
- Jenny Rose Finkel and Christopher D Manning. 2008. Enforcing transitivity in coreference resolution. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies*, pages 45–48.

- J Fukumoto, F Masui, M Shimohata, and M Sasaki. 1997. OKI electric industry: Description of the OKI system as used for MUC-7. In *Proceedings of the 7th Message Understanding Conference*, pages 1–7.
- Roberto Garigliano, Agnieszka Urbanowicz, and David J Nettleton. 1998. Description of the LOLITA system as used in MUC-7. In *Proceedings of the 7th Message Understanding Conference*, pages 71–85.
- Ralph Grishman and Beth Sundheim. 1996. Message understanding conference-6: A brief history. In *Proceedings of the 6th Message Understanding Conference*, pages 466–471.
- Barbara J Grosz and Candace L Sidner. 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.
- Barbara J Grosz, Scott Weinstein, and Aravind K Joshi. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–225.
- Liliane Haegeman. 1991. *Introduction to government and binding theory*. Blackwell Publishing.
- Aria Haghighi 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*, pages 1152–1161.
- Hannaneh Hajishirzi, Leila Zilles, Daniel S Weld, and Luke Zettlemoyer. 2013. Joint coreference resolution and named entity linking with multi-pass sieves. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 289–299.
- Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th Conference on Computational Linguistics*, pages 539–545.

- Irene Heim. 1982. *The semantics of definite and indefinite noun phrases*. Ph.D. thesis, University of Massachusetts, Amherst, Massachusetts, USA.
- Lynette Hirschman, Patricia Robinson, John Burger, and Marc Vilain. 1997. Automating coreference: The role of annotated training data. In *Proceedings of the Association for the Advancement of Artificial Intelligence Spring Symposium on Applying Machine Learning to Discourse Processing*, pages 118–121.
- Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: the 90% solution. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 57–60.
- Kevin Humphreys, Robert Gaizauskas, Saliha Azzam, Chris Huyck, Brian Mitchell, Hamish Cunningham, and Yorick Wilks. 1998. University of Sheffield: Description of the LaSIE-II system as used for MUC-7. In *Proceedings of the 7th Message Understanding Conference*.
- Joseph Irwin, Mamoru Komachi, and Yuji Matsumoto. 2011. Narrative schema as world knowledge for coreference resolution. In *Proceedings of the 15th Conference on Computational Natural Language Learning: Shared Task*, pages 86–92.
- Jay J Jiang and David W Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of ROCLING X*, pages 19–33.
- Andrew Kehler and Hannah Rohde. 2013. A probabilistic reconciliation of coherence-driven and centering-driven theories of pronoun interpretation. *Theoretical Linguistics*, 39(1-2):1–37.
- Laura Kertz, Andrew Kehler, and Jeff Elman. 2006. Grammatical and coherence-based factors in pronoun interpretation. In *Proceedings of the 28th annual conference of the Cognitive Science Society*, pages 1605–1610.

Paul Kingsbury and Martha Palmer. 2002. From treebank to propbank. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, pages 1989–1993.

Jonathan K Kummerfeld and Dan Klein. 2013. Error-driven analysis of challenges in coreference resolution. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 265–277.

Claudia Leacock and Martin Chodorow. 1998. Combining local context and WordNet similarity for word sense identification. *WordNet: An electronic lexical database*, pages 265–283.

Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2011. Stanford’s multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Proceedings of the 15th Conference on Computational Natural Language Learning: Shared Task*, pages 28–34.

Hector J Levesque. 2011. The winograd schema challenge. In *Proceedings of the CommonSense-11 Symposium*, pages 63–68.

Dekang Lin. 1998a. Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 768–774.

Dekang Lin. 1998b. An information-theoretic definition of similarity. In *International Conference on Machine Learning*, volume 98, pages 296–304.

Dekang Lin. 1998c. Using collocation statistics in information extraction. In *Proceedings of the 7th Message Understanding Conference*.

Linguistic Data Consortium. 2008. ACE (Automatic Content Extraction) English Annotation Guidelines for Entities.

<https://www ldc.upenn.edu/sites/www ldc.upenn.edu/files/english-entities-guidelines-v6.6.pdf>.

Linguistic Data Consortium. 2008. *2008 ACE: Cross-Document Annotation Guidelines (XDOC)*. <https://www ldc.upenn.edu/sites/www ldc.upenn.edu/files/ace08-xdoc-1.6.pdf>.

Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In *Proceedings of the 2005 Joint Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 25–32.

Xiaoqiang Luo, Abe Ittycheriah, Hongyan Jing, Nanda Kambhatla, and Salim Roukos. 2004. A mention-synchronous coreference resolution algorithm based on the Bell tree. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pages 135–142.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Ruslan Mitkov. 1999. Anaphora resolution: the state of the art. Technical report, University of Wolverhampton, Wolverhampton, England.

Jennifer Mooney and Ian Jolliffe. 2003. Sensitivity of the  $\chi^2$  goodness-of-fit test to the choice of classes. *Teaching Statistics*, 26:22–23.

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

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

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

National Institute of Standards and Technology. 2005. The ace 2005 (ACE05) evaluation plan. evaluation of the detection and recognition of ACE entities, values, temporal expressions, relations, and events.

Karl Pearson. 1895. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, pages 240–242.

Haoruo Peng, Daniel Khashabi, and Dan Roth. 2015. Solving hard coreference problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 809–819.

Simone Paolo Ponzetto and Michael Strube. 2006. Exploiting semantic role labeling, WordNet and Wikipedia for coreference resolution. In *Proceedings of the 2006 Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 192–199.

Sameer Pradhan, Xiaoqiang Luo, Marta Recasens, Eduard Hovy, Vincent Ng, and Michael Strube. 2014. Scoring coreference partitions of predicted mentions: A reference implementation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: Short Papers*, pages 30–35.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Conference on Natural Language Learning: Shared Task*, pages 1–40.

Sameer Pradhan, Lance Ramshaw, Mitchell Marcus, Martha Palmer, Ralph Weischedel, and Nianwen Xue. 2011. CoNLL-2011 shared task: Modeling unrestricted coreference

- in ontonotes. In *Proceedings of the 15th Conference on Computational Natural Language Learning: Shared Task*, pages 1–27.
- Sameer Pradhan, Wayne Ward, Kadri Hacioglu, James Martin, and Dan Jurafsky. 2004. Shallow semantic parsing using support vector machines. In *Proceedings of the Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 141–144.
- Roy Rada, Hamed Mili, Ellen Bicknell, and Maria Blettner. 1989. Development and application of a metric on semantic nets. *Institute of Electrical and Electronics Engineers Transactions on Systems, Man and Cybernetics*, 19(1):17–30.
- Karthik Raghunathan, Heeyoung Lee, Sudarshan Rangarajan, Nathanael Chambers, Mihai Surdeanu, Dan Jurafsky, and Christopher Manning. 2010. A multi-pass sieve for coreference resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 492–501.
- Altaf Rahman and Vincent Ng. 2009. Supervised models for coreference resolution. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 968–977.
- Altaf Rahman and Vincent Ng. 2011. Coreference resolution with world knowledge. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 814–824.
- Altaf Rahman and Vincent Ng. 2012. Resolving complex cases of definite pronouns: the Winograd schema challenge. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 777–789.
- William M Rand. 1971. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850.

- Keith Rayner. 1998. Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin*, 124(3):372.
- Marta Recasens, Marie-Catherine de Marneffe, and Christopher Potts. 2013. The life and death of discourse entities: Identifying singleton mentions. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 627–633.
- Marta Recasens and Eduard Hovy. 2011. BLANC: Implementing the rand index for coreference evaluation. *Natural Language Engineering*, 17(04):485–510.
- Marta Recasens, Eduard Hovy, and M Antònia Martí. 2011. Identity, non-identity, and near-identity: Addressing the complexity of coreference. *Lingua*, 121(6):1138–1152.
- Philip Resnik. 1995. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 448–453.
- Hannah Rohde, Andrew Kehler, and Jeffrey L Elman. 2007. Pronoun interpretation as a side effect of discourse coherence. In *Proceedings of the 29th annual conference of the cognitive science society*, pages 617–622.
- Bertrand Russell. 1905. On denoting. *Mind*, pages 479–493.
- Roger C Schank and Robert P Abelson. 1977. *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. Psychology Press.
- Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521–544.
- Charles Spearman. 1904. The proof and measurement of association between two things. *The American journal of psychology*, 15(1):72–101.

- Veselin Stoyanov, Uday Babbar, Pracheer Gupta, and Claire Cardie. 2011. Reconciling OntoNotes: Unrestricted coreference resolution in OntoNotes with Reconcile. In *Proceedings of the 15th Conference on Computational Natural Language Learning: Shared Task*, pages 122–126.
- Veselin Stoyanov, Claire Cardie, Nathan Gilbert, Ellen Riloff, David Buttler, and David Hysom. 2010a. Coreference resolution with reconcile. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 156–161.
- Veselin Stoyanov, Claire Cardie, Nathan Gilbert, Ellen Riloff, David Buttler, and David Hysom. 2010b. Reconcile: A coreference resolution research platform. Technical report, Cornell University Tech Reports, Ithaca, New York, USA.
- Veselin Stoyanov, 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 Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, pages 656–664.
- Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: A core of semantic knowledge. In *Proceedings of the 16th International World Wide Web Conference*, pages 697–706.
- Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394.
- Alan M Turing. 1950. Computing machinery and intelligence. *Mind*, pages 433–460.
- Olga Uryupina. 2003. High-precision identification of discourse new and unique noun phrases. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 80–86.

- Olga Uryupina, Massimo Poesio, Claudio Giuliano, and Kateryna Tymoshenko. 2011. Disambiguation and filtering methods in using web knowledge for coreference resolution. In *Proceedings of the 24th International Florida Artificial Intelligence Research Society Conference*, pages 317–322.
- Kees Van Deemter and Rodger Kibble. 2000. On coreferring: Coreference in MUC and related annotation schemes. *Computational Linguistics*, 26(4):629–637.
- Yannick Versley. 2008. Vagueness and referential ambiguity in a large-scale annotated corpus. *Research on Language and Computation*, 6(3-4):333–353.
- Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. In *Proceedings of the 6th conference on Message understanding*, pages 45–52.
- Kellie Webster and James R Curran. 2014. Limited memory incremental coreference resolution. In *Proceedings of the 25th International Conference on Computational Linguistics*, pages 2129–2139.
- Kellie Webster and Joel Nothman. 2016. Using mention accessibility to improve coreference resolution. In *Proceedings of the 54th Annual Conference of the Association of Computational Linguistics*.
- Sam Wiseman, Alexander M. Rush, Stuart M. Shieber, and Jason Weston. 2015. Learning anaphoricity and antecedent ranking features for coreference resolution. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, volume Volume 1: Long Papers, pages 1416–1426.
- Zhibiao Wu and Martha Palmer. 1994. Verbs semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics*, pages 133–138.

Xiaofeng Yang, Jian Su, Guodong Zhou, and Chew Lim Tan. 2004. An NP-cluster based approach to coreference resolution. In *Proceedings of the 20th International Conference on Computational Linguistics*, pages 226–232.

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