Unrestricted Coreference: Identifying Entities and Events in OntoNotes

Sameer S. Pradhan, Lance Ramshaw, Ralph Weischedel, Jessica MacBride and Linnea Micciulla
BBN Technologies
Cambridge, MA 02138

Abstract

Most research in the field of anaphora or coreference detection has been limited to noun phrase coreference, usually on a restricted set of entities, such as ACE entities. In part, this has been due to the lack of corpus resources tagged with general anaphoric coreference. The OntoNotes project is creating a large-scale, accurate corpus for general anaphoric coreference that covers entities and events not limited to noun phrases or a limited set of entity types. The coreference layer in OntoNotes constitutes one part of a multi-layer, integrated annotation of shallow semantic structure in text. This paper presents an initial model for unrestricted coreference based on this data that uses a machine learning architecture with state-of-the-art features. Significant improvements can be expected from using such cross-layer information for training predictive models. This paper describes the coreference annotation in OntoNotes, presents the baseline model, and provides an analysis of the contribution of this new resource in the context of recent MUC and ACE results.

1 Introduction

The importance of the coreference resolution or entity/event detection task, namely identifying all mentions of entities in text and clustering them into equivalence classes, has been well recognized in the natural language processing community. Researchers have applied machine learning techniques to identify entities in text [17, 10]. Improved learning techniques have been developed recently to push the performance forward [2, 6, 1], and various different knowledge sources from shallow semantics to encyclopedic knowledge are being exploited [14, 15, 18, 9].

Corpora to support this task date back to the Message Understanding Conferences (MUC-6, MUC-7) [8, 3]. These corpora were tagged with coreferring entities identified by noun phrases in the text. However, the amount of data tagged was considerably smaller and less consistent (in terms of inter-annotator agreement [4]) than one would hope to have in order to get good statistical evidence in the form of lexical coverage and semantic relatedness that could be used by a classifier to generate better predictive models. The importance of a well-defined tagging scheme, and measuring consistency has been recognized and studied in the past [12, 13, 11].

More recently, the Automatic Content Extraction (ACE) program, a successor to MUC, has generated corpora for coreference modeling, although the focus in this program shifted from general anaphoric coreference to working with a limited set of entities. Over this same period, significant improvements have been made in the field of language processing in general, and various machine learning techniques have been developed that can achieve good performance on these tasks. However, there is a growing consensus that in order for these to be most useful for language understanding applications such as question answering or the newly coined distillation task—both of which seek to take information access technology to the next level—we need larger, more consistent, and wider coverage automatic entity and event identification techniques.

The OntoNotes project addresses this bottleneck by tagging general anaphoric coreference, which not only considers entities described using noun phrases, but also encompasses events described through verbs, and by targeting a larger collection of text and a degree of consistency that should be beneficial to an automatic learning architecture. Another significant feature of this corpus is that the coreference annotation is situated among multiple other annotation layers including syntax and propositional structure – for both verbs and nouns, name entities and word senses which will be connected to concepts in an ontology [5] – each of which are annotated with a high degree of consistency and in three languages: English, Chinese and Arabic.

In this paper, we briefly describe the type of annotation in the OntoNotes coreference layer, provide a baseline performance measure for unrestricted coreference on English OntoNotes data using a state-of-the-art feature set, and analyze the results of that model.

Corpora	Documents		
	Total	Train	Test
OntoNotes	597	484	113
MUC-6	60	30	30
MUC-7	50	30	20
ACE 2003	521	422	97
ACE 2005	535	-	-

Table 1. Number of documents in the OntoNotes data, and some comparison with the MUC and ACE data sets

2 Corpus

In this section we will discuss the data that is being tagged in the OntoNotes project and the details of the coreference annotation.

2.1 Data

The English portion of the OntoNotes Year 1 corpus comprises 300k words, in a 597 document collection from the Wall Street Journal (WSJ) newswire, annotated with about 11400 coreference chains. This is text that has been Treebanked and PropBanked in the past. The number of documents in this corpus are shown in Table 1. For purpose of comparison we have also listed the number of documents in the MUC-6, MUC-7, and ACE corpora. The MUC-6 data was also taken from WSJ, whereas MUC-7 data was from the New York Times. A portion of the ACE data was newswire articles from various sources (159/522 for year 2003 and 124/535 for year 2005), while the rest was from other genres.

2.2 Annotation

Two different types of coreference are distinguished in the OntoNotes data: Identical (IDENT), and Appositive (APPOS). (Appositives are treated separately because they function as attributions, as described further below.) The IDENT type is used for anaphoric coreference, meaning links between pronominal, nominal, and named mentions of specific referents. It does not include mentions of generic, underspecified, or abstract entities. The total coreference chains in the data are composed of about 9300 IDENT chains and 2100 APPOS chains.

Coreference is annotated for all specific entities and events. There is no limit on the semantic types of NP entities that can be considered for coreference, and in particular, coreference is not limited to ACE types.

The mentions over which IDENT coreference applies are typically pronominal, named, or definite nominal. The annotation process begins by automatically extracting all of the NP mentions from the Penn Treebank, though the annotators can also add additional mentions when appropriate. In the following two examples (and later ones), the phrases notated in bold form the links of an IDENT chain.

(1) She had **a good suggestion** and **it** was unanimously accepted by all.

(2) **Elco Industries Inc.** said **it** expects net income in the year ending June 30, 1990, to fall below a recent analyst's estimate of $ 1.65 a share. **The Rockford, Ill. maker of fasteners** also said **it** expects to post sales in the current fiscal year that are "slightly above" fiscal 1989 sales of $ 155 million.

2.2.1 Verbs

Verbs are added as single-word spans if they can be coreferenced with a noun phrase or with another verb. The intent is to annotate the VP, but we mark the single-word head for convenience. This includes morphologically related nominalizations (3) and noun phrases that refer to the same event, even if they are lexically distinct from the verb (4). In the following two examples, only the chains related to the *growth* event are shown.

(3) Sales of passenger cars **grew** 22%. **The strong growth** followed year-to-year increases.

(4) Japan's domestic sales of cars, trucks and buses in October **rose** 18% from a year earlier to 500,004 units, a record for the month, the Japan Automobile Dealers' Association said. The strong **growth** followed year-to-year increases of 21% in August and 12% in September.

2.2.2 Pronouns

All pronouns and demonstratives are linked to anything that they refer to, and pronouns in quoted speech are also marked. Expletive or pleonastic pronouns (*it*, *there*) are not considered for tagging, and generic *you* is not marked. In the following example, the pronoun *you* and *it* would not be marked. (In this and following examples, an asterisk (*) before a boldface phrase identifies entity/event mentions that would *not* be tagged as coreferent.)

(5) Senate majority leader Bill Frist likes to tell a story from his days as a pioneering heart surgeon back in Tennessee. A lot of times, Frist recalls, ***you'd** have a critical patient lying there waiting for a new heart, and ***you'd** want to cut, but ***you** couldn't start unless ***you** knew that the replacement heart would make ***it** to the operating room.

2.2.3 Generic mentions

Generic nominal mentions can be linked with referring pronouns and other definite mentions, but are not linked to generic nominal mentions. This would allow linking of the bracketed mentions in (6) and (7), but not (8).

(6) **Officials** said **they** are tired of making the same statements.

(7) **Meetings** are most productive when **they** are held in the morning. **Those meetings**, however, generally have the worst attendance.

(8) Allergan Inc. said it received approval to sell the PhacoFlex intraocular lens, the first foldable silicone lens available for ***cataract surgery**. The lens' foldability enables it to be inserted in smaller incisions than are now possible for ***cataract surgery**.

Bare plurals, as in (6) and (7), are always considered generic. In example (9) below, there are two generic instances of *parents*. These are marked as distinct IDENT chains (with separate chains distinguished by subscripts X, Y and Z), each containing a generic and the related referring pronouns.

(9) **Parents**$_X$ should be involved with **their**$_X$ children's education at home, not in school. **They**$_X$ should see to it that **their**$_X$ kids don't play truant; **they**$_X$ should make certain that the children spend enough time doing homework; **they**$_X$ should scrutinize the report card. **Parents**$_Y$ are too likely to blame schools for the educational limitations of **their**$_Y$ children. If **parents**$_Z$ are dissatisfied with a school, **they**$_Z$ should have the option of switching to another.

In (10) below, the verb "halve" cannot be linked to "a reduction of 50%", since "a reduction" is indefinite.

(10) Argentina said it will ask creditor banks to ***halve** its foreign debt of $64 billion – the third-highest in the developing world . Argentina aspires to reach ***a reduction of 50%** in the value of its external debt.

2.2.4 Premodifiers

Proper premodifiers can be coreferenced, but proper nouns that are in a morphologically adjectival form are treated as adjectives, and not coreferenced. For example, adjectival forms of GPEs such as *Chinese* in "the Chinese leader", would not be linked. Thus we could coreference *United States* in "the United States policy" with another referent, but not *American* "the American policy." GPEs and Nationality acronyms (e.g. *U.S.S.R.* or *U.S.*). are also considered adjectival. Pre-modifier acronyms can be coreferenced unless they refer to a nationality. Thus in the examples below, *FBI* can be coreferenced to other mentions, but *U.S.* cannot.

(11) **FBI** spokesman

(12) ***U.S.** spokesman

Dates and monetary amounts can be considered part of a coreference chain even when they occur as premodifiers.

(13) The current account deficit on France's balance of payments narrowed to 1.48 billion French francs ($236.8 million) in August from a revised 2.1 billion francs in **July**, the Finance Ministry said. Previously, the **July** figure was estimated at a deficit of 613 million francs.

(14) The company's **$150** offer was unexpected. The firm balked at **the price**.

2.2.5 Copular verbs

Attributes signaled by copular structures are not marked; these are attributes of the referent they modify, and their relationship to that referent will be captured through word sense and propositional argument tagging.

(15) **John**$_X$ is a linguist. **People**$_Y$ are nervous around **John**$_X$, because **he**$_X$ always corrects **their**$_Y$ grammar.

Copular (or 'linking') verbs are those verbs that function as a copula and are followed by a subject complement. Some common copular verbs are: *be, appear, feel, look, seem, remain, stay, become, end up, get*. Subject complements following such verbs are considered attributes, and not linked. Since *Called* is copular, neither IDENT nor APPOS coreference is marked in the following case.

(16) Called Otto's Original Oat Bran Beer, the brew costs about $12.75 a case.

2.2.6 Small clauses

Like copulas, small clause constructions are not marked. The following example is treated as if the copula were present ("John considers Fred to be an idiot"):

(17) John considers ***Fred *an idiot**.

2.2.7 Temporal expressions

Temporal expressions such as the following are linked:

(18) John spent **three years** in jail. In **that time**...

Deictic expressions such as *now, then, today, tomorrow, yesterday,* etc. can be linked, as well as other temporal expressions that are relative to the time of the writing of the article, and which may therefore require knowledge of the time of the writing to resolve the coreference. Annotators were allowed to use knowledge from outside the text in resolving these cases. In the following example, *the end of this period* and *that time* can be coreferenced, as can *this period* and *from three years to seven years*.

(19) The limit could range from three years to seven years, depending on the composition of the management team and the nature of its strategic plan. At the end of this period, the poison pill would be eliminated automatically, unless a new poison pill were approved by the then-current shareholders, who would have an opportunity to evaluate the corporation's strategy and management team at that time.

In multi-date temporal expressions, embedded dates are not separately connected to to other mentions of that date. For example in *Nov. 2, 1999*, *Nov.* would not be linked to another instance of *November* later in the text.

2.2.8 Appositives

Because they logically represent attributions, appositives are tagged separately from Identity coreference. They consist of a head, or referent (a noun phrase that points to a specific object/concept in the world), and one or more attributes of that referent. An appositive construction contains a noun phrase that modifies an immediately-adjacent noun phrase (separated only by a comma, colon, dash, or parenthesis). It often serves to rename or further define the first mention. Marking appositive constructions allows us to capture the attributed property even though there is no explicit copula.

(20) **John**$_{\text{head}}$, **a linguist**$_{\text{attribute}}$

The head of each appositive construction is distinguished from the attribute according to the following heuristic specificity scale:

Proper noun	>Pronoun	>Def. NP	>Indef. spec.	>Non-spec. NP
John	>He	>the man	>a man I know	>man

This leads to the following cases:

(21) **John**$_{\text{head}}$, **a linguist**$_{\text{attribute}}$

(22) **A famous linguist**$_{\text{attribute}}$, **he**$_{\text{head}}$ studied at ...

(23) **a principal of the firm**$_{\text{attribute}}$, **J. Smith**$_{\text{head}}$

In cases where the two members of the appositive are equivalent in specificity, the left-most member of the appositive is marked as the head/referent. Definite NPs include NPs with a definite marker (*the*) as well as NPs with a possessive adjective (his). Thus the first element is the head in all of the following cases:

(24) The chairman, the man who never gives up

(25) The sheriff, his friend

(26) His friend, the sheriff

In the specificity scale, specific names of diseases and technologies are classified as proper names, whether they are capitalized or not.

(27) A dangerous bacteria, bacillium, is found

When the entity to which an appositive refers is also mentioned elsewhere, only the single span containing the entire appositive construction is included in the larger IDENT chain. None of the nested NP spans are linked. In the example below, the entire span can be linked to later mentions to *Richard Godown*. The sub-spans are not included separately in the IDENT chain.

(28) **Richard Godown, president of the Industrial Biotechnology Association**

Ages are tagged as attributes (as if they were ellipses of, for example, *a 42-year-old*):

(29) **Mr.Smith**$_{\text{head}}$, **42**$_{\text{attribute}}$,

2.2.9 Special Issues

In addition to the ones above, there are some special cases such as:

- No coreference is marked between an organization and its members.
- GPEs are linked to references to their governments, even when the references are nested NPs, or the modifier and head of a single NP.

3 Serif

Our baseline model for full coreference has its roots in the coreference component of the Serif system [16]. Serif is BBN's information extraction system which has primarily been designed to address various information extraction tasks in the ACE program. In this section we will describe the part of the Serif architecture that is relevant in the context of coreference decoding. Given a portion of text to decode, it carries out the following sequence of analysis on it:

- Tokenization
- Part of Speech Tagging
- Name Recognition
- Syntactic Parsing
- Mention Detection
- Nominal Classification
- Entity Linker

The first four steps are fairly self-explanatory. The name recognizer used is BBN's IdentiFinder™. It was trained on a mixture of TDT-4 and WSJ data. (The WSJ training was restricted to Sections 02-21.) IdentiFinder performance typically ranges in the mid-90s for ACE entities. The parser is a generative syntactic parser trained on Sections 02-21 of the Penn Treebank. It is worth mentioning that the parser constrains the resulting phrase structure so as not to clash with the names identified in the previous step. The parser performance ranges in the high-80s. In the mention detection step, all the possible phrases in the text that could be potential entity mentions are identified. Then, in the nominal classification step, each of the nominal mentions thus tagged is classified either as one of the ACE entity types or as as of an OTHER type. Coreference processing happens during the entity linker stage, when the related mentions are clustered together into entity sets.

The following sections describe the coreference decoding algorithm used in Serif in more detail.

3.1 Coreference Algorithm

The coreference algorithm is based on a layered architecture, as is common in the field. Coreference is performed by making a left to right pass over the document. For each sentence, proper name mentions are linked first, followed by nominal mentions, and then by pronouns. All three layers of linking are completed for one sentence before moving on to the next one. Each decision point is a decision about whether to link the current mention to an already existing entity, or to create a new entity. In subsequent sentences, entities mentioned in previous sentences are also considered as possible referents.

3.1.1 Name Coreference

This module links name mentions of the same type together, using a generative model based on various features including string match and edit distance, along with acronym and abbreviation heuristics.

3.1.2 Nominal Coreference

By this stage, all the nominal mentions in the text have been classified as one of the different ACE types. This module uses a log linear classifier with perceptron-trained weights, trained for 10 epochs. During training the model uses the "closest-search" strategy. For each training mention, the model traverses the previous mentions in reverse order until it reaches a correct match, at which point it stops making comparisons. Then in the decoding pass, the model creates similar features and accepts the highest scoring decision. The search is restricted to mentions of the same entity type as identified in the earlier nominal classification phase.

The model uses a standard set of lexical and syntactic features similar to those used in other recent machine learning work on coreference resolution [17, 10], such as distance, gender, number, syntactic head words, WordNet class, etc. One unusual, additional feature involves bit strings based on a word cluster tree of the type described in [7] that is automatically derived from bigram statistics taken from a large corpus of unannotated text. In this representation, a word is associated with various-length bit vector signatures. Shorter vectors represent more general and longer vectors more specific word clusters. The model assigns weight to the particular length signatures found to provide useful generalization during training.

3.1.3 Pronoun Coreference

The version of the pronoun linker used for the experiments in this paper is a log-linear model with perceptron-trained feature weights, similar to the nominal coreference model just described.

3.2 Performance

Serif is a state-of-the-art ACE system. Its performance on the task of Entity Detection in the ACE 2007 evaluation is shown in Table 2. The ACE score represents a weighted measure over all the different ACE entity types. Full results from that evaluation with a detailed analysis and explanation of the metrics used can be found at the following URL: `http://www.nist.gov/speech/tests/ace/ace07/-doc/ace07_eval_official_results_20070402.htm`

Genre	ACE Score
Broadcast Conversation	44.7
Broadcast News	65.4
Newswire	58.1
Telephone	49.2
Usenet	39.2
Weblogs	52.7
Overall	56.3

Table 2. Performance of the Serif system in the ACE-2007 Evaluation.

4 Serif++

This section describes our extended model for full coreference, trained on OntoNotes data. The OntoNotes annotation is stored as a relational database, providing an integrated repository for the multiple levels of structural information that OntoNotes provides. This database is accompanied by an object-oriented API for storing and manipulating this information, and we plan to use it not only as a repository for the layers of semantic information, but also as a means of extracting novel cross-layer features for improving, and facilitating a level of joint estimation over all these layers. In order to facilitate that, we have abstracted the modules that provide the foundation for the task of coreference decoding, and have started building another system, which we call Serif++. Our current version of this system uses an algorithm that parallels the fundamental Serif coreference algorithm. It departs from the core algorithm in the following ways: i) it trains separate models for the "Other" category; ii) it uses a Support Vector Machine classifier; iii) it allows verbs to be connected to nominalized entities, as happens in the event coreference portion of OntoNotes.

4.1 Extending the scope of Entities

Since there is no previous coreference baseline on the OntoNotes test set, we compare our performance here to Serif, as an existing state-of-the-art coreference decoding system. However, Serif only tags ACE type entities. In order to get some indication of the performance that a system might achieve if the coreference is not limited to only ACE types, we made a small alteration to the Serif algorithm, utilizing a built-in backoff mechanism, so that all the nominal mentions that were not tagged as one of the many ACE entities were classified into an "OTHER" class. All the mentions of this class were then only allowed to link to other mentions of that class. Pronouns, on the other hand, were allowed to link either to "OTHER" entities or to standard ACE entities. Using these modifications, new models were trained using all of the ACE 2004 data (train and test portion), ACE 2005 train set (both adjudicated and non-adjudicated annotations), ACE 2007 training data, and a small amount of data translated from Arabic and Chinese for the ET-2007 evaluation. (None of this data overlapped with any part of the OntoNotes corpus.)

Since the ACE corpus, collected over the years, has much more total annotated data (1500 documents) than OntoNotes, we also trained this model on a subset of 484 documents (the same number of documents forming the OntoNotes training collection, Sections 02-21) to provide an additional point of comparison on a comparably-sized corpus. These 484 documents were composed of files from the ACE 2005 and 2007 training data. In addition, we tuned the linking threshold used by the nominal linker on the OntoNotes development Section 24.

4.2 Training Serif on OntoNotes data

We also wanted to compare the Serif++ baseline system performance with that of Serif itself, and in order to do that, we also retrained the original Serif model on the OntoNotes data. Owing to the structure and assumptions underlying Serif, this process required some approximations, but it provides a useful metric for comparison.

In order to reformat the OntoNotes data to match the format required by Serif, we used the named entity information layer which is another part of OntoNotes, and re-mapped those richer name types to ACE types. Types that were not part of the ACE were mapped to the "OTHER" category. Also, as part of OntoNotes annotation guidelines, only the larger of nested NPs with the same head are identified as entities. However, tagging these as non-coreferent would confuse the head word statistics of Serif, so in order to avoid this problem, we automatically tagged all the nested NPs with the same head as belonging to the same coreference chain. In a post-processing step, we resolved all the nested entities automatically, keeping only the longest spanning links. During testing, we used the automatically generated parse and name entity information generated by Serif.

	Train	System	All F	ACE Name F	Ace Nominals F	Other Nominals F	Pronoun F
Test	OntoNotes	Serif++	51.2	60.5	41.2	21.9	54.5
(Section 23)	OntoNotes	Serif	48.8	61.4	44.4	20.0	56.2
	ACE	Serif	48.9	62.1	47.8	10.1	58.1
	ACE (484 documents)	Serif	46.7	62.8	46.3	9.5	55.0

Table 3. Performance on the task of IDENT **recognition on OntoNotes development and test sets.**

5 Experiments

In this section we will report results of our system on the OntoNotes data. We use Section 23 of the OntoNotes data as our test set and Section 24 as the development set.

5.1 Results

For scoring purposes, we used the MUC scoring software [20]. In the OntoNotes data, heads of coreference links are not explicitly tagged, as is the case with the MUC corpus. For the purposes of this evaluation, we decided not to use the "MIN" attribute in the MUC data, which gives full credit to links in a chain that have the head word as part of its span as long as the entire span does not exceed the span of the link identified in the annotated data [3]. In other words, only extents that exactly match the extents in the tagged data received full credit.

5.1.1 Identity Coreference (IDENT)

Table 3 shows the performance of four different configurations on the OntoNotes development and test sets. The column "All" lists performance over all the different entity and events. "ACE Name"and "ACE Nominals" refer to entities of one of the known ACE name types. (To identify this category, we use the OntoNotes portion tagged with nominals belonging to known ACE entities.) "Other Nominals" were other nominals, tagged as "OTHER". (Unavoidably some of these are also proper name entities that do not belong to the ACE types, so, this category should be considered as a hybrid between "ACE Name" and "ACE Nominals"). "Pronoun" covers pronominal mentions. Performance of the systems on each of these categories is listed in the columns with the respective headings. Several interesting observations can be made: i) The difference in performance between the configuration where Serif is trained using 484 documents and one that uses much larger ACE training data (of 1500 documents), is not very large. ii) The best performance is exhibited by Serif++, which is a bit higher than the one that is trained on modified training data. This comparison is limited and not completely fair, since the OntoNotes data had to be reformatted and squeezed in the Serif mold, and since Serif is tuned to the type of coreference that is described by the ACE guidelines, it does not allow verbs to be linked to their nominal mentions, and it does not differentiate between Appositive coreference and Identification coreference. iii) The performance on the "Other Nominals" category is somewhat more worse than the "ACE Nominals" category, and both of them are the lowest scoring categories in this set. Since one of the most important feature harvested by the learning mechanisms is the head word of the nominal mentions, this tends to indicate that there is still a significant lack of important semantic information and world-knowledge incorporation which prevents the learning mechanism from making accurate connections between phrases headed by different head words (also known as coreference bridging [19]) and make distinctions between the ones that do share the same head words, but describe different entities. Some errors we looked at confirmed this hypothesis; iv) Finally and perhaps most important, the performance on "Other Nominals" when training systems on the OntoNotes data is much higher than without that data. We experimented with several threshold parameters in the nominal linking model of Serif to check whether relaxing the thresholds would tend to identify between "Other Nominals" entities, but it turned out that the number of false positives that get generated far outweigh the improvement in recall. Although the "Other Nominals" performance in this initial test is still quite poor, the breadth of feature information in this corpus should enable performance to be improved.

5.1.2 Appositive Coreference (APPOS)

Since Serif does not distinguish between Identity and Appositive coreference, we only report performance on this category for Serif++. This model uses a fairly simple set of features based on phrase types, name entities, punctuation before and after the phrases, and syntactic paths between the constituents. The performance of this model on the tasks of identifying appositions and classifying the components as either "Head" or "Attribute" is shown in Table 4. For English in particular, there is a distinct surface structure given

	System	Id. F	Classification F
Test (Section 23)	Serif++	87.3	93.5

Table 4. Performance on the task of APPOS **identification and classification on the OntoNotes development and test sets.**

to appositions, and so the performance is much higher.

6 Conclusions

In this paper we described the general anaphoric coreference information tagged in the OntoNotes corpus, and presented a baseline system for learning such unrestricted entities and events in text. This annotation has been done as a layer among several other semantic annotation layers such as syntax, propositions and word senses. From the results it is clear that data tagged with more general entities and events does help recover significantly more rich information from text, improving coreference performance for entities of known types. However, the baseline model's performance for nominal entities of unknown types is quite low, indicating that richer semantic information will need to be incorporated along with improvements in learning mechanisms to achieve high performance on this more difficult kind of unrestricted coreference

7 Acknowledgments

We gratefully acknowledge the support of the Defense Advanced Research Projects Agency (DARPA/IPTO) under the GALE program, DARPA/CMO Contract No. HR0011-06-C-0022.

References

[1] P. Denis and J. Baldridge. Joint determination of anaphoricity and coreference resolution using integer programming. In *Proceedings of HLT/NAACL*, 2007.

[2] S. M. Harabagiu, R. C. Bunescu, and S. J. Maiorano. Text and knowledge mining for coreference resolution. In *Proceedings of NAACL*, 2001.

[3] L. Hirschman and N. Chinchor. Coreference task definition (v3.0, 13 jul 97). In *Proceedings of the Seventh Message Understanding Conference*.

[4] L. Hirschman, P. Robinson, J. Burger, and M. Vilain. Automating coreference: The role of annotated training data. In *Proceedings of AAAI Spring Symposium on Applying Machine Learning to Discourse Processing*, 1998.

[5] E. Hovy, M. Marcus, M. Palmer, L. Ramshaw, and R. Weischedel. OntoNotes: The 90% solution. In *Proceedings of HLT/NAACL*, 2006.

[6] A. McCallum and B. Wellner. Conditional models of identity uncertainty with application to noun coreference. In *Advances in Neural Information Processing Systems (NIPS)*, 2004.

[7] S. Miller, J. Guinness, and A. Zamanian. Name tagging with word clusters and distriminative training. In *Proceedings of HLT/NAACL*, 2004.

[8] MUC-6. Coreference task definition (v2.3, 8 sep 95). In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, 1995.

[9] V. Ng. Shallow semantics for coreference resolution. In *Proceedings of IJCAI*, 2007.

[10] V. Ng and C. Cardie. Improving machine learning approaches to coreference resolution. In *Proceedings of ACL*, 2002.

[11] R. Passonneau. Computing reliability for coreference annotation. In *Proceedings of LREC*, 2004.

[12] M. Poesio. The mate/gnome scheme for anaphoric annotation, revisited. In *Proceedings of SIGDIAL*, 2004.

[13] M. Poesio and R. Artstein. The reliability of anaphoric annotation, reconsidered: Taking ambiguity into account. In *Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky*, 2005.

[14] S. Ponzetto and M. Strube. Semantic role labeling for coreference resolution. In *Proceedings of EACL*, Italy, April 2005.

[15] S. Ponzetto and M. Strube. Exploiting semantic role labeling, WordNet and Wikipedia for coreference resolution. In *Proceedings of HLT/NAACL*, pages 192–199, New York City, N.Y., June 2006.

[16] L. Ramshaw and R. Weischedel. Information extraction. In *Proceedings of IEEE ICASSP*, Philadelphia, PA, 2005.

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

[18] Y. Versley. Antecedent selection techniques for high-recall coreference resolution. In *Proceedings of EMNLP/CoNLL*, 2007.

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

[20] M. Vilain, J. Burger, J. Aberdeen, D. Connolly, and L. Hirschman. A model theoretic coreference scoring scheme. In *Proceedings of Message Undersatnding Conference (MUC-6)*.