A Multi-Pass Sieve for Coreference Resolution

Karthik Raghunathan, Heeyoung Lee, Sudarshan Rangarajan, Nathanael Chambers,
Mihai Surdeanu, Dan Jurafsky, Christopher Manning
Computer Science Department
Stanford University, Stanford, CA 94305
{kr,heeyoung,sudarshn,natec,mihais,jurafsky,manning}@stanford.edu

Abstract

Most coreference resolution models determine if two mentions are coreferent using a single function over a set of constraints or features. This approach can lead to incorrect decisions as lower precision features often overwhelm the smaller number of high precision ones. To overcome this problem, we propose a simple coreference architecture based on a sieve that applies tiers of deterministic coreference models one at a time from highest to lowest precision. Each tier builds on the previous tier's entity cluster output. Further, our model propagates global information by sharing attributes (e.g., gender and number) across mentions in the same cluster. This cautious sieve guarantees that stronger features are given precedence over weaker ones and that each decision is made using all of the information available at the time. The framework is highly modular: new coreference modules can be plugged in without any change to the other modules. In spite of its simplicity, our approach outperforms many state-of-the-art supervised and unsupervised models on several standard corpora. This suggests that sieve-based approaches could be applied to other NLP tasks.

1 Introduction

Recent work on coreference resolution has shown that a rich feature space that models lexical, syntactic, semantic, and discourse phenomena is crucial to successfully address the task (Bengston and Roth, 2008; Haghighi and Klein, 2009; Haghighi and Klein, 2010). When such a rich representation is available, even a simple deterministic model can achieve state-of-the-art performance (Haghighi and Klein, 2009).

By and large most approaches decide if two mentions are coreferent using a single function over all these features and information local to the two mentions.[1] This is problematic for two reasons: (1) lower precision features may overwhelm the smaller number of high precision ones, and (2) local information is often insufficient to make an informed decision. Consider this example:

The second attack occurred after some rocket firings aimed, apparently, toward [the israelis], apparently in retaliation. [we]'re checking our facts on that one. ... the president, quoted by ari fleischer, his spokesman, is saying he's concerned the strike will undermine efforts by palestinian authorities to bring an end to terrorist attacks and does not contribute to the security of [israel].

Most state-of-the-art models will incorrectly link *we* to *the israelis* because of their proximity and compatibility of attributes (both *we* and *the israelis* are plural). In contrast, a more cautious approach is to first cluster *the israelis* with *israel* because the demonymy relation is highly precise. This initial clustering step will assign the correct animacy attribute (`inanimate`) to the corresponding geo-political entity, which will prevent the incorrect merging with the mention *we* (`animate`) in later steps.

We propose an unsupervised sieve-like approach to coreference resolution that addresses these is-

[1] As we will discuss below, some approaches use an additional component to infer the overall best mention clusters for a document, but this is still based on confidence scores assigned using local information.

Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pages 492–501,
MIT, Massachusetts, USA, 9-11 October 2010.

sues. The approach applies tiers of coreference models one at a time from highest to lowest precision. Each tier builds on the entity clusters constructed by previous models in the sieve, guaranteeing that stronger features are given precedence over weaker ones. Furthermore, each model's decisions are richly informed by sharing attributes across the mentions clustered in earlier tiers. This ensures that each decision uses all of the information available at the time. We implemented all components in our approach using only deterministic models. All our components are unsupervised, in the sense that they do not require training on gold coreference links.

The contributions of this work are the following:

- We show that a simple scaffolding framework that deploys strong features through tiers of models performs significantly better than a single-pass model. Additionally, we propose several simple, yet powerful, new features.
- We demonstrate how far one can get with simple, deterministic coreference systems that do not require machine learning or detailed semantic information. Our approach outperforms most other unsupervised coreference models and several supervised ones on several datasets.
- Our modular framework can be easily extended with arbitrary models, including statistical or supervised models. We believe that our approach also serves as an ideal platform for the development of future coreference systems.

2 Related Work

This work builds upon the recent observation that strong features outweigh complex models for coreference resolution, in both supervised and unsupervised learning setups (Bengston and Roth, 2008; Haghighi and Klein, 2009). Our work reinforces this observation, and extends it by proposing a novel architecture that: (a) allows easy deployment of such features, and (b) infuses global information that can be readily exploited by these features or constraints.

Most coreference resolution approaches perform the task by aggregating local decisions about pairs of mentions (Bengston and Roth, 2008; Finkel and Manning, 2008; Haghighi and Klein, 2009; Stoyanov, 2010). Two recent works that diverge from this pattern are Culotta et al. (2007) and Poon and Domingos (2008). They perform coreference resolution jointly for all mentions in a document, using first-order probabilistic models in either supervised or unsupervised settings. Haghighi and Klein (2010) propose a generative approach that models entity clusters explicitly using a mostly-unsupervised generative model. As previously mentioned, our work is not constrained by first-order or Bayesian formalisms in how it uses cluster information. Additionally, the deterministic models in our tiered model are significantly simpler, yet perform generally better than the complex inference models proposed in these works.

From a high level perspective, this work falls under the theory of shaping, defined as a "method of successive approximations" for learning (Skinner, 1938). This theory is known by different names in many NLP applications: Brown et al. (1993) used simple models as "stepping stones" for more complex word alignment models; Collins (1999) used "cautious" decision list learning for named entity classification; Spitkovsky et al. (2010) used "baby steps" for unsupervised dependency parsing, etc. To the best of our knowledge, we are the first to apply this theory to coreference resolution.

3 Description of the Task

Intra-document coreference resolution clusters together textual mentions within a single document based on the underlying referent entity. Mentions are usually noun phrases (NPs) headed by nominal or pronominal terminals. To facilitate comparison with most of the recent previous work, we report results using gold mention boundaries. However, our approach does not make any assumptions about the underlying mentions, so it is trivial to adapt it to predicted mention boundaries (e.g., see Haghighi and Klein (2010) for a simple mention detection model).

3.1 Corpora

We used the following corpora for development and evaluation:

- **ACE2004-ROTH-DEV**[2] – development split of Bengston and Roth (2008), from the corpus used in the 2004 Automatic Content Extraction (ACE) evaluation. It contains 68 documents and 4,536 mentions.

[2]We use the same corpus names as (Haghighi and Klein, 2009) to facilitate comparison with previous work.

- **ACE2004-CULOTTA-TEST** – partition of ACE 2004 corpus reserved for testing by several previous works (Culotta et al., 2007; Bengston and Roth, 2008; Haghighi and Klein, 2009). It consists of 107 documents and 5,469 mentions.
- **ACE2004-NWIRE** – the newswire subset of the ACE 2004 corpus, utilized by Poon and Domingos (2008) and Haghighi and Klein (2009) for testing. It contains 128 documents and 11,413 mentions.
- **MUC6-TEST** – test corpus from the sixth Message Understanding Conference (MUC-6) evaluation. It contains 30 documents and 2,068 mentions.

We used the first corpus (ACE2004-ROTH-DEV) for development. The other corpora are reserved for testing. We parse all documents using the Stanford parser (Klein and Manning, 2003). The syntactic information is used to identify the mention head words and to define the ordering of mentions in a given sentence (detailed in the next section). For a fair comparison with previous work, we do not use gold named entity labels or mention types but, instead, take the labels provided by the Stanford named entity recognizer (NER) (Finkel et al., 2005).

3.2 Evaluation Metrics

We use three evaluation metrics widely used in the literature: (a) pairwise F1 (Ghosh, 2003) – computed over mention pairs in the same entity cluster; (b) MUC (Vilain et al., 1995) – which measures how many predicted clusters need to be merged to cover the gold clusters; and (c) B^3 (Amit and Baldwin, 1998) – which uses the intersection between predicted and gold clusters for a given mention to mark correct mentions and the sizes of the the predicted and gold clusters as denominators for precision and recall, respectively. We refer the interested reader to (X. Luo, 2005; Finkel and Manning, 2008) for an analysis of these metrics.

4 Description of the Multi-Pass Sieve

Our sieve framework is implemented as a succession of independent coreference models. We first describe how each model selects candidate mentions, and then describe the models themselves.

4.1 Mention Processing

Given a mention m_i, each model may either decline to propose a solution (in the hope that one of the subsequent models will solve it) or deterministically select a single best antecedent from a list of previous mentions $m_1, \ldots, m_{i-1}$. We sort candidate antecedents using syntactic information provided by the Stanford parser, as follows:

Same Sentence – Candidates in the same sentence are sorted using left-to-right breadth-first traversal of syntactic trees (Hobbs, 1977). Figure 1 shows an example of candidate ordering based on this traversal. The left-to-right ordering favors subjects, which tend to appear closer to the beginning of the sentence and are more probable antecedents. The breadth-first traversal promotes syntactic salience by ranking higher noun phrases that are closer to the top of the parse tree (Haghighi and Klein, 2009). If the sentence containing the anaphoric mention contains multiple clauses, we repeat the above heuristic separately in each `S*` constituent, starting with the one containing the mention.

Previous Sentence – For all nominal mentions we sort candidates in the previous sentences using right-to-left breadth-first traversal. This guarantees syntactic salience and also favors document proximity. For pronominal mentions, we sort candidates in previous sentences using left-to-right traversal in order to favor subjects. Subjects are more probable antecedents for pronouns (Kertz et al., 2006). For example, this ordering favors the correct candidate (*pepsi*) for the mention *they*:

> *[pepsi] says it expects to double [quaker]'s snack food growth rate. after a month-long courtship, [they] agreed to buy quaker oats...*

In a significant departure from previous work, each model in our framework gets (possibly incomplete) clustering information for each mention from the earlier coreference models in the multi-pass system. In other words, each mention m_i may already be assigned to a cluster C_j containing a set of mentions: $C_j = \{m_1^j, \ldots, m_k^j\}$; $m_i \in C_j$. Unassigned mentions are unique members of their own cluster. We use this information in several ways:

Attribute sharing – Pronominal coreference resolution (discussed later in this section) is severely af-

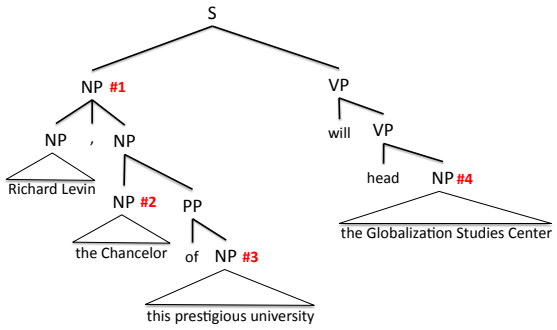

Figure 1: Example of left-to-right breadth-first tree traversal. The numbers indicate the order in which the NPs are visited.

fected by missing attributes (which introduce precision errors because incorrect antecedents are selected due to missing information) and incorrect attributes (which introduce recall errors because correct links are not generated due to attribute mismatch between mention and antecedent). To address this issue, we perform a union of all mention attributes (e.g., number, gender, animacy) in a given cluster and share the result with all cluster mentions. If attributes from different mentions contradict each other we maintain all variants. For example, our naive number detection assigns `singular` to the mention *a group of students* and `plural` to *five students*. When these mentions end up in the same cluster, the resulting number attributes becomes the set {`singular`, `plural`}. Thus this cluster can later be merged with both singular and plural pronouns.

Mention selection – Traditionally, a coreference model attempts to resolve every mention in the text, which increases the likelihood of errors. Instead, in each of our models, we exploit the cluster information received from the previous stages by resolving only mentions that are currently first in textual order in their cluster. For example, given the following ordered list of mentions, $\{m_1^1, m_2^2, m_3^2, m_4^3, m_5^1, m_6^2\}$, where the superscript indicates cluster id, our model will attempt to resolve only m_2^2 and m_4^3. These two are the only mentions that have potential antecedents and are currently marked as the first mentions in their clusters. The intuition behind this heuristic is two-fold. First, early cluster mentions are usually better defined than subsequent ones, which are likely to have fewer modifiers or are pronouns (Fox, 1993). Several of our models use this modifier information. Second, by definition, first mentions appear closer to the beginning of the document, hence there are fewer antecedent candidates to select from, and fewer opportunities to make a mistake.

Search Pruning – Finally, we prune the search space using *discourse salience*. We disable coreference for first cluster mentions that: (a) are or start with indefinite pronouns (e.g., *some*, *other*), or (b) start with indefinite articles (e.g., *a*, *an*). One exception to this rule is the model deployed in the first pass; it only links mentions if their entire extents match exactly. This model is triggered for all nominal mentions regardless of discourse salience, because it is possible that indefinite mentions are repeated in a document when concepts are discussed but not instantiated, e.g., *a sports bar* below:

Hanlon, a longtime Broncos fan, thinks it is the perfect place for [a sports bar] and has put up a blue-and-orange sign reading, "Wanted Broncos Sports Bar On This Site." . . . In a Nov. 28 letter, Proper states "while we have no objection to your advertising the property as a location for [a sports bar], using the Broncos' name and colors gives the false impression that the bar is or can be affiliated with the Broncos."

4.2 The Modules of the Multi-Pass Sieve

We now describe the coreference models implemented in the sieve. For clarity, we summarize them in Table 1 and show the cumulative performance as they are added to the sieve in Table 2.

4.2.1 Pass 1 - Exact Match

This model links two mentions only if they contain exactly the same extent text, including modifiers and determiners, e.g., *the Shahab 3 ground-ground missile*. As expected, this model is extremely precise, with a pairwise precision over 96%.

4.2.2 Pass 2 - Precise Constructs

This model links two mentions if any of the conditions below are satisfied:

Appositive – the two nominal mentions are in an appositive construction, e.g., *[Israel's Deputy Defense Minister], [Ephraim Sneh] , said . . .* We use the same syntactic rules to detect appositions as Haghighi and Klein (2009).

Pass	Type	Features
1	N	exact extent match
2	N,P	appositive | predicate nominative | role appositive | relative pronoun | acronym | demonym
3	N	cluster head match & word inclusion & compatible modifiers only & not i-within-i
4	N	cluster head match & word inclusion & not i-within-i
5	N	cluster head match & compatible modifiers only & not i-within-i
6	N	relaxed cluster head match & word inclusion & not i-within-i
7	P	pronoun match

Table 1: Summary of passes implemented in the sieve. The Type column indicates the type of coreference in each pass: N – nominal or P – pronominal. & and | indicate conjunction and disjunction of features, respectively.

Predicate nominative – the two mentions (nominal or pronominal) are in a copulative subject-object relation, e.g., *[The New York-based College Board] is [a nonprofit organization that administers the SATs and promotes higher education]* (Poon and Domingos, 2008).

Role appositive – the candidate antecedent is headed by a noun and appears as a modifier in an NP whose head is the current mention, e.g., *[[actress] Rebecca Schaeffer]*. This feature is inspired by Haghighi and Klein (2009), who triggered it only if the mention is labeled as a person by the NER. We constrain this heuristic more in our work: we allow this feature to match only if: (a) the mention is labeled as a person, (b) the antecedent is animate (we detail animacy detection in Pass 7), and (c) the antecedent's gender is not neutral.

Relative pronoun – the mention is a relative pronoun that modifies the head of the antecedent NP, e.g., *[the finance street [which] has already formed in the Waitan district]*.

Acronym – both mentions are tagged as NNP and one of them is an acronym of the other, e.g., *[Agence France Presse] ... [AFP]*. We use a simple acronym detection algorithm, which marks a mention as an acronym of another if its text equals the sequence of upper case characters in the other mention. We will adopt better solutions for acronym detection in future work (Schwartz, 2003).

Demonym – one of the mentions is a demonym of the other, e.g., *[Israel] ... [Israeli]*. For demonym detection we use a static list of countries and their gentilic forms from Wikipedia.[3]

All the above features are extremely precise. As shown in Table 2 the pairwise precision of the sieve after adding these features is over 95% and recall increases 5 points.

4.2.3 Pass 3 - Strict Head Matching

Linking a mention to an antecedent based on the naive matching of their head words generates a lot of spurious links because it completely ignores possibly incompatible modifiers (Elsner and Charniak, 2010). For example, *Yale University* and *Harvard University* have similar head words, but they are obviously different entities. To address this issue, this pass implements several features that must all be matched in order to yield a link:

Cluster head match – the mention head word matches *any* head word in the antecedent cluster. Note that this feature is actually more relaxed than naive head matching between mention and antecedent candidate because it is satisfied when the mention's head matches the head of any entity in the candidate's cluster. We constrain this feature by enforcing a conjunction with the features below.

Word inclusion – all the non-stop[4] words in the mention cluster are included in the set of non-stop words in the cluster of the antecedent candidate. This heuristic exploits the property of discourse that it is uncommon to introduce novel information in later mentions (Fox, 1993). Typically, mentions of the same entity become shorter and less informative as the narrative progresses. For example, the two mentions in *... intervene in the [Florida Supreme Court]'s move ... does look like very dramatic change made by [the Florida court]* point to the same entity, but the two mentions in the text below belong to different clusters:

The pilot had confirmed ... he had turned onto

[3] http://en.wikipedia.org/wiki/List_of_adjectival_and_demonymic_forms_of_place_names

[4] Our stop word list includes person titles as well.

	MUC			B^3			Pairwise		
Passes	P	R	F1	P	R	F1	P	R	F1
{1}	95.9	31.8	47.8	99.1	53.4	69.4	96.9	15.4	26.6
{1,2}	95.4	43.7	59.9	98.5	58.4	73.3	95.7	20.6	33.8
{1,2,3}	92.1	51.3	65.9	96.7	62.9	76.3	91.5	26.8	41.5
{1,2,3,4}	91.7	51.9	66.3	96.5	63.5	76.6	91.4	27.8	42.7
{1,2,3,4,5}	91.1	52.6	66.7	96.1	63.9	76.7	90.3	28.4	43.2
{1,2,3,4,5,6}	89.5	53.6	67.1	95.3	64.5	76.9	88.8	29.2	43.9
{1,2,3,4,5,6,7}	83.7	74.1	78.6	88.1	74.2	80.5	80.1	51.0	62.3

Table 2: Cumulative performance on development (ACE2004-ROTH-DEV) as passes are added to the sieve.

[the correct runway] but pilots behind him say he turned onto [the wrong runway].

Compatible modifiers only – the mention's modifiers are all included in the modifiers of the antecedent candidate. This feature models the same discourse property as the previous feature, but it focuses on the two individual mentions to be linked, rather than their entire clusters. For this feature we only use modifiers that are nouns or adjectives.

Not i-within-i – the two mentions are not in an i-within-i construct, i.e., one cannot be a child NP in the other's NP constituent (Haghighi and Klein, 2009).

This pass continues to maintain high precision (91% pairwise) while improving recall significantly (over 6 points pairwise and almost 8 points MUC).

4.2.4 Passes 4 and 5 - Variants of Strict Head

Passes 4 and 5 are different relaxations of the feature conjunction introduced in Pass 3, i.e., Pass 4 removes the `compatible modifiers only` feature, while Pass 5 removes the `word inclusion` constraint. All in all, these two passes yield an improvement of 1.7 pairwise F1 points, due to recall improvements. Table 2 shows that the `word inclusion` feature is more precise than `compatible modifiers only`, but the latter has better recall.

4.2.5 Pass 6 - Relaxed Head Matching

This pass relaxes the cluster head match heuristic by allowing the mention head to match any word in the cluster of the candidate antecedent. For example, this heuristic matches the mention *Sanders* to a cluster containing the mentions {*Sauls*, *the judge*, *Circuit Judge N. Sanders Sauls*}. To maintain high precision, this pass requires that both mention and antecedent be labeled as named entities and the types coincide. Furthermore, this pass implements a conjunction of the above features with `word inclusion` and `not i-within-i`. This pass yields less than 1 point improvement in most metrics.

4.2.6 Pass 7 - Pronouns

With one exception (Pass 2), all the previous coreference models focus on nominal coreference resolution. However, it would be incorrect to say that our framework ignores pronominal coreference in the first six passes. In fact, the previous models prepare the stage for pronominal coreference by constructing precise clusters with shared mention attributes. These are crucial factors for pronominal coreference.

Like previous work, we implement pronominal coreference resolution by enforcing agreement constraints between the coreferent mentions. We use the following attributes for these constraints:

Number – we assign number attributes based on: (a) a static list for pronouns; (b) NER labels: mentions marked as a named entity are considered singular with the exception of organizations, which can be both singular or plural; (c) part of speech tags: `NN*S` tags are plural and all other `NN*` tags are singular; and (d) a static dictionary from (Bergsma and Lin, 2006).

Gender – we assign gender attributes from static lexicons from (Bergsma and Lin, 2006; Ji and Lin, 2009).

Person – we assign person attributes only to pronouns. However, we do not enforce this constraint when linking two pronouns if one appears within quotes. This is a simple heuristic for speaker detection, e.g., *I* and *she* point to the same person in

"[I] voted my conscience," [she] said.

Animacy – we set animacy attributes using: (a) a static list for pronouns; (b) NER labels, e.g., `PERSON` is animate whereas `LOCATION` is not; and (c) a dictionary boostrapped from the web (Ji and Lin, 2009).

NER label – from the Stanford NER.

If we cannot detect a value, we set attributes to `unknown` and treat them as wildcards, i.e., they can match any other value.

This final model raises the pairwise recall of our system almost 22 percentage points, with only an 8 point drop in pairwise precision. Table 2 shows that similar behavior is measured for all other metrics. After all passes have run, we take the transitive closure of the generated clusters as the system output.

5 Experimental Results

We present the results of our approach and other relevant prior work in Table 3. We include in the table all recent systems that report results under the same conditions as our experimental setup (i.e., using gold mentions) and use the same corpora. We exclude from this analysis two notable works that report results only on a version of the task that includes finding mentions (Haghighi and Klein, 2010; Stoyanov, 2010). The Haghighi and Klein (2009) numbers have two variants: with semantics (+S) and without (−S). To measure the contribution of our multi-pass system, we also present results from a single-pass variant of our system that uses all applicable features from the multi-pass system (marked as "single pass" in the table).

Our sieve model outperforms all systems on two out of the four evaluation corpora (ACE2004-ROTH-DEV and ACE2004-NWIRE), on all metrics. On the corpora where our model is not best, it ranks a close second. For example, in ACE2004-CULOTTA-TEST our system has a B^3 F1 score only .4 points lower than Bengston and Roth (2008) and it outperforms all unsupervised approaches. In MUC6-TEST, our sieve's B^3 F1 score is 1.8 points lower than Haghighi and Klein (2009) +S, but it outperforms a supervised system that used gold named entity labels. Finally, the multi-pass architecture always beats the equivalent single-pass system with its contribution ranging between 1 and 4 F1 points depending on the corpus and evaluation metric.

Our approach has the highest precision on all corpora, regardless of evaluation metric. We believe this is particularly useful for large-scale NLP applications that use coreference resolution components, e.g., question answering or information extraction. These applications can generally function without coreference information so it is beneficial to provide such information only when it is highly precise.

6 Discussion

6.1 Comparison to Previous Work

The sieve model outperforms all other systems on at least two test sets, even though most of the other models are significantly richer. Amongst the comparisons, several are supervised (Bengston and Roth, 2008; Finkel and Manning, 2008; Culotta et al., 2007). The system of Haghighi and Klein (2009) +S uses a lexicon of semantically-compatible noun pairs acquired transductively, i.e., with knowledge of the mentions in the test set. Our system does not rely on labeled corpora for training (like supervised approaches) nor access to corpora during testing (like Haghighi and Klein (2009)).

The system that is closest to ours is Haghighi and Klein (2009) −S. Like us, they use a rich set of features and deterministic decisions. However, theirs is a single-pass model with a smaller feature set (no cluster-level, acronym, demonym, or animacy information). Table 3 shows that on the two corpora where results for this system are available, we outperform it considerably on all metrics. To understand if the difference is due to the multi-pass architecture or the richer feature set we compared (Haghighi and Klein, 2009) −S against both our multi-pass system and its single-pass variant. The comparison indicates that both these contributions help: our single-pass system outperforms Haghighi and Klein (2009) consistently, and the multi-pass architecture further improves the performance of our single-pass system between 1 and 4 F1 points, depending on the corpus and evaluation metric.

6.2 Semantic Head Matching

Recent unsupervised coreference work from Haghighi and Klein (2009) included a novel semantic component that matched related head words (e.g., *AOL* is a *company*) learned from select

	MUC			B^3			Pairwise		
	P	R	F1	P	R	F1	P	R	F1
ACE2004-ROTH-DEV									
This work (sieve)	**83.7**	**74.1**	**78.6**	**88.1**	74.2	**80.5**	**80.1**	51.0	**62.3**
This work (single pass)	82.2	72.6	77.1	86.8	72.6	79.1	76.0	47.6	58.5
Haghighi and Klein (2009) **–S**	78.3	70.5	74.2	84.0	71.0	76.9	71.3	45.4	55.5
Haghighi and Klein (2009) **+S**	77.9	**74.1**	75.9	81.8	**74.3**	77.9	68.2	**51.2**	58.5
ACE2004-CULOTTA-TEST									
This work (sieve)	**80.4**	71.8	75.8	**86.3**	75.4	**80.4**	**71.6**	46.2	56.1
This work (single pass)	78.4	69.2	73.5	85.1	73.9	79.1	69.5	44.1	53.9
Haghighi and Klein (2009) **–S**	74.3	66.4	70.2	83.6	71.0	76.8	66.4	38.0	48.3
Haghighi and Klein (2009) **+S**	74.8	**77.7**	**79.6**	79.6	**78.5**	79.0	57.5	**57.6**	**57.5**
Culotta et al. (2007)	–	–	–	86.7	73.2	79.3	–	–	–
Bengston and Roth (2008)	82.7	69.9	75.8	**88.3**	**74.5**	**80.8**	55.4	63.7	59.2
MUC6-TEST									
This work (sieve)	**90.5**	68.0	77.7	**91.2**	61.2	73.2	**90.3**	53.3	67.1
This work (single pass)	89.3	65.9	75.8	90.2	58.8	71.1	89.5	50.6	64.7
Haghighi and Klein (2009) **+S**	87.2	**77.3**	**81.9**	84.7	**67.3**	**75.0**	80.5	**57.8**	**67.3**
Poon and Domingos (2008)	83.0	75.8	79.2	–	–	–	63.0	57.0	60.0
Finkel and Manning (2008) **+G**	89.7	55.1	68.3	90.9	49.7	64.3	74.1	37.1	49.5
ACE2004-NWIRE									
This work (sieve)	**83.8**	73.2	**78.1**	**87.5**	71.9	**78.9**	**79.6**	46.2	**58.4**
This work (single pass)	82.2	71.5	76.5	86.2	70.0	77.3	76.9	41.9	54.2
Haghighi and Klein (2009) **+S**	77.0	**75.9**	76.5	79.4	**74.5**	76.9	66.9	**49.2**	56.7
Poon and Domingos (2008)	71.3	70.5	70.9	–	–	–	62.6	38.9	48.0
Finkel and Manning (2008) **+G**	78.7	58.5	67.1	86.8	65.2	74.5	76.1	44.2	55.9

Table 3: Results using gold mention boundaries. Where available, we show results for a given corpus grouped in two blocks: the top block shows results of unsupervised systems and the bottom block contains supervised systems. Bold numbers indicate best results in a given block. **+/-S** indicates if the (Haghighi and Klein, 2009) system includes/excludes their semantic component. **+G** marks systems that used gold NER labels.

wikipedia articles. They first identified articles relevant to the entity mentions in the test set, and then bootstrapped from known syntactic patterns for apposition and predicate-nominatives in order to learn a database of related head pairs. They show impressive gains by using these learned pairs in coreference decisions. This type of learning using test set mentions is often described as *transductive*.

Our work instead focuses on an approach that does not require access to the dataset beforehand. We thus did not include a similar semantic component in our system, given that running a bootstrapping learner whenever a new data set is encountered is not practical and, ultimately, reduces the usability of this NLP component. However, our results show that our sieve algorithm with minimal semantic information still performs as well as the Haghighi and Klein (2009) system with semantics.

6.3 Flexible Architecture

The sieve architecture offers benefits beyond improved accuracy. Its modular design provides a flexibility for features that is not available in most supervised or unsupervised systems. The sieve allows new features to be seamlessly inserted without affecting (or even understanding) the other components. For instance, once a new high precision feature (or group of features) is inserted as its own stage, it will benefit later stages with more precise clusters, but it will not interfere with their particu-

lar algorithmic decisions. This flexibility is in sharp contrast to supervised classifiers that require their models to be retrained on labeled data, and unsupervised systems that do not offer a clear insertion point for new features. It can be difficult to fully understand how a system makes a single decision, but the sieve allows for flexible usage with minimal effort.

6.4 Error Analysis

	Pronominal	Nominal	Proper	Total
Pronominal	49 / 237	116 / 317	104 / 595	269 / 1149
Nominal	79 / 351	129 / 913	61 / 986	269 / 2250
Proper	51 / 518	15 / 730	38 / 595	104 / 1843
Total	179 / 1106	260 / 1960	203 / 2176	642 / 5242

Table 4: Number of pair-wise errors produced by the sieve after transitive closure in the MUC6-TEST corpus. Rows indicate mention types; columns are types of antecedent. Each cell shows the number of precision/recall errors for that configuration. The total number of gold links in MUC6-TEST is 11,236.

Table 4 shows the number of incorrect pair-wise links generated by our system on the MUC6-TEST corpus. The table indicates that most of our errors are for nominal mentions. For example, the combined (precision plus recall) number of errors for proper or common noun mentions is three times larger than the number of errors made for pronominal mentions. The table also highlights that most of our errors are recall errors. There are eight times more recall errors than precision errors in our output. This is a consequence of our decision to prioritize highly precise features in the sieve.

The above analysis illustrates that our next effort should focus on improving recall. In order to understand the limitations of our current system, we randomly selected 60 recall errors (20 for each mention type) and investigated their causes. Not surprisingly, the causes are unique to each type.

For proper nouns, 50% of recall errors are due to *mention lengthening*, mentions that are longer than their earlier mentions. For example, *Washington-based USAir* appears after *USAir* in the text, so our head matching components skip it because their high precision depends on disallowing new modifiers as the discourse proceeds. When the mentions were reversed (as is the usual case), they match.

The common noun recall errors are very different from proper nouns: 17 of the 20 random examples can be classified as *semantic knowledge*. These errors are roughly evenly split between recognizing categories of names (e.g., Gitano is an organization name hence it should match the nominal antecedent *the company*), and understanding hypernym relations like *settlements* and *agreements*.

Pronoun errors come in two forms. Roughly 40% of these errors are attribute mismatches involving sometimes ambiguous uses of gender and number (e.g., *she* with *Pat Carney*). Another 40% are not semantic or attribute-based, but rather simply arise due to the order in which we check potential antecedents. In all these situations, the correct links are missed because the system chooses a closer (incorrect) antecedent.

These four highlighted errors (lengthening, semantics, attributes, ordering) add up to 77% of all recall errors in the selected set. In general, each error type is particular to a specific mention type. This suggests that recall improvements can be made by focusing on one mention type without aversely affecting the others. Our sieve-based approach to coreference uniquely allows for such new models to be seamlessly inserted.

7 Conclusion

We presented a simple deterministic approach to coreference resolution that incorporates document-level information, which is typically exploited only by more complex, joint learning models. Our sieve architecture applies a battery of deterministic coreference models one at a time from highest to lowest precision, where each model builds on the previous model's cluster output. Despite its simplicity, our approach outperforms or performs comparably to the state of the art on several corpora.

An additional benefit of the sieve framework is its modularity: new features or models can be inserted in the system with limited understanding of the other features already deployed. Our code is publicly released[5] and can be used both as a stand-alone coreference system and as a platform for the development of future systems.

[5] http://nlp.stanford.edu/software/dcoref.shtml

The strong performance of our system suggests the use of sieves in other NLP tasks for which a variety of very high-precision features can be designed and non-local features can be shared; likely candidates include relation and event extraction, template slot filling, and author name deduplication.

Acknowledgments

We gratefully acknowledge the support of the Defense Advanced Research Projects Agency (DARPA) Machine Reading Program under Air Force Research Laboratory (AFRL) prime contract no. FA8750-09-C-0181. Any opinions, findings, and conclusion or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of DARPA, AFRL, or the US government.

Many thanks to Jenny Finkel for writing a reimplementation of much of Haghighi and Klein (2009), which served as the starting point for the work reported here. We also thank Nicholas Rizzolo and Dan Roth for helping us replicate their experimental setup, and Heng Ji and Dekang Lin for providing their gender lexicon.

References

B. Amit and B. Baldwin. 1998. Algorithms for scoring coreference chains. In *MUC-7*.

E. Bengston and D. Roth. 2008. Understanding the value of features for coreference resolution. In *EMNLP*.

S. Bergsma and D. Lin. 2006. Bootstrapping Path-Based Pronoun Resolution. In *ACL-COLING*.

P.F. Brown, V.J. Della Pietra, S.A. Della Pietra, and R.L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19.

M. Collins and Y. Singer. 1999. Unsupervised models for named entity classification. In *EMNLP-VLC*.

A. Culotta, M. Wick, R. Hall, and A. McCallum. 2007. First-order probabilistic models for coreference resolution. In *NAACL-HLT*.

M. Elsner and E. Charniak. 2010. The same-head heuristic for coreference. In *ACL*.

J. Finkel, T. Grenager, and C. Manning. 2005. Incorporating non-local information into information extraction systems by Gibbs sampling. In *ACL*.

J. Finkel and C. Manning. 2008. Enforcing transitivity in coreference resolution. In *ACL*.

B. A. Fox 1993. *Discourse structure and anaphora: written and conversational English*. Cambridge University Press.

J. Ghosh. 2003. Scalable clustering methods for data mining. *Handbook of Data Mining*, chapter 10, pages 247–277.

A. Haghighi and D. Klein. 2009. Simple coreference resolution with rich syntactic and semantic features. In *EMNLP*.

A. Haghighi and D. Klein. 2010. Coreference resolution in a modular, entity-centered model. In *HLT-NAACL*.

J.R. Hobbs. 1977. Resolving pronoun references. *Lingua*.

H. Ji and D. Lin. 2009. Gender and animacy knowledge discovery from web-scale n-grams for unsupervised person mention detection. In *PACLIC*.

L. Kertz, A. Kehler, and J. Elman. 2006. Grammatical and Coherence-Based Factors in Pronoun Interpretation. In *Proceedings of the 28th Annual Conference of the Cognitive Science Society*.

D. Klein and C. Manning. 2003. Accurate unlexicalized parsing. In *ACL*.

X. Luo. 2005. On coreference resolution performance metrics. In *HTL-EMNLP*.

H. Poon and P. Domingos. 2008. Joint unsupervised coreference resolution with Markov Logic. In *EMNLP*.

A.S. Schwartz and M.A. Hearst. 2003. A simple algorithm for identifying abbrevation definitions in biomedical text. In *Pacific Symposium on Biocomputing*.

B.F. Skinner. 1938. *The behavior of organisms: An experimental analysis*. Appleton-Century-Crofts.

V.I. Spitkovsky, H. Alshawi, and D. Jurafsky. 2010. From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing. In *NAACL*.

V. Stoyanov, N. Gilbert, C. Cardie, and E. Riloff. 2010. Conundrums in noun phrase coreference resolution: making sense of the state-of-the-art. In *ACL-IJCNLP*.

M. Vilain, J. Burger, J. Aberdeen, D. Connolly, and L. Hirschman. 1995. A model-theoretic coreference scoring scheme. In *MUC-6*.