Learning-based Multi-Sieve Co-reference Resolution with Knowledge*

Lev Ratinov
Google Inc.†
ratinov@google.com

Dan Roth
University of Illinois at Urbana-Champaign
danr@illinois.edu

Abstract

We explore the interplay of knowledge and structure in co-reference resolution. To inject knowledge, we use a state-of-the-art system which cross-links (or "grounds") expressions in free text to Wikipedia. We explore ways of using the resulting grounding to boost the performance of a state-of-the-art co-reference resolution system. To maximize the utility of the injected knowledge, we deploy a learning-based multi-sieve approach and develop novel entity-based features. Our end system outperforms the state-of-the-art baseline by 2 B^3 F1 points on non-transcript portion of the ACE 2004 dataset.

1 Introduction

Co-reference resolution is the task of grouping mentions to entities. For example, consider the text snippet in Fig. 1[1]. The correct output groups the mentions $\{m_1, m_2, m_5\}$ to one entity while leaving m_3 and m_4 as singletons. Resolving co-reference is fundamental for understanding natural language. For example in Fig. 1, to infer that Kusrk has suffered a torpedo detonation, we have to understand that $\{[\textit{vessel}]\}_{m1}$ refers to $\{[\textit{Kursk}]\}_{m2}$.

"After the $\{[\textit{vessel}]\}_{m_1}$ *suffered a catastrophic torpedo detonation,* $\{[\textit{Kursk}]\}_{m_2}$ *sank in the waters of* $\{[\textit{Barents Sea}]\}_{m_3}$ *with all hands lost. Though rescue attempts were offered by a nearby* $\{\textit{Norwegian}\ [\textit{ship}]\}_{m_4}$*, Russia declined initial rescue offers, and all 118 sailors and officers aboard* $\{[\textit{Kursk}]\}_{m_5}$ *perished."*

Figure 1: Example illustrating the challenges in co-reference resolution.

This inference is typically trivial for humans, but proves extremely challenging for state-of-the-art co-reference resolution systems. We believe that it is world knowledge that gives people the ability to understand text with such ease. A human reader can infer that since Kursk sank, it must be a vessel and vessels which suffer catastrophic torpedo detonations can sink. Moreover, some readers might *just know* that *Kursk* is a Russian submarine named after the city of Kursk, where the largest tank battle in history took place in 1943. In this work we are using Wikipedia as a source of encyclopedic knowledge. The key contributions of this work are:

(1) Using Wikipedia to assign a set of knowledge attributes to mentions in a context-sensitive way. For example, for the text in Fig. 1, our system assigns to the mention *"Kursk"* the nationalities: *Russian, Soviet* and the attributes *ship, incident, submarine, shipwreck* (as opposed to *city* or *battle*). We are using a publicly available system for context-

* We thank Nicholas Rizzolo and Kai Wei Chang for their invaluable help with modifying the baseline co-reference system. We thank the anonymous EMNLP reviewers for constructive comments. This research was supported by the Army Research Laboratory (ARL) under agreement W911NF-09-2-0053 and by 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, conclusions or recommendations are those of the authors and do not necessarily reflect the view of the ARL, DARPA, AFRL, or the US government.

† The majority of this work was done while the first author was at the University of Illinois.

[1] Throughout this paper, curly brackets {} denote the extent and square brackets [] denote the head.

sensitive disambiguation to Wikipedia. We then extract attributes from the cross-linked Wikipedia pages (described in Sec. 3.1), assign these attributes to the document mentions (Sec. 3.2) and develop knowledge-rich compatibility metric between mentions (Sec. 3.3)[2].

(2) Integrating the strength of rule-based systems such as (Haghighi and Klein, 2009; Raghunathan et al., 2010) into a machine learning framework. We are using a multi-sieve approach (Raghunathan et al., 2010), which splits pairwise "co-reference" vs. "non-coreference" decisions to different types and attempts to make the easy decisions first (Goldberg and Elhadad, 2010). Our multi-sieve approach is different from (Raghunathan et al., 2010) in several respects: (a) our sieves are machine-learning classifiers, (b) the same pair of mentions can fall into multiple sieves, (c) later sieves can override the decisions made by earlier sieves, allowing to recover from errors as additional evidence becomes available. In our running example, the decision of whether $\{[\textit{vessel}]\}_{m1}$ refers to $\{[\textit{Kursk}]\}_{m2}$ is made before the decision of whether $\{[\textit{vessel}]\}_{m1}$ refers to $\{\textit{Norwegian } [\textit{ship}]\}_{m_4}$ since decisions in the same sentence are believed to be easier than cross-sentence ones. We describe our learning-based multi-sieve approach in Sec. 4.

(3) A novel approach for entity-based features. As sieves of classifiers are applied, our system attempts to model entities and share the attributes between the mentions belonging to the same entity. Once the decision that $\{[\textit{vessel}]\}_{m1}$ and $\{[\textit{Kursk}]\}_{m2}$ co-refer is made, we want the two mentions to share the ***Russian*** nationality. This allows us to avoid erroneously linking $\{[\textit{vessel}]\}_{m1}$ to $\{\textit{Norwegian } [\textit{ship}]\}_{m_4}$ despite *vessel* and *ship* being synonyms in WordNet. However, in this work we allow the sieves to make conflicting decisions on the same pair of mentions. Hence, obtaining entities and their attributes by straightforward transitive closure of co-reference predictions is impossible. We describe our approach for leveraging possibly contradicting predictions in Sec. 5.

(4) By adding word-knowledge features and using learning-based multi-sieve approach, we improve the performance of the state-of-the-art system of (Bengtson and Roth, 2008) by 3 MUC, 2 B^3 and 2 CEAF F1 points on the non-transcript portion of the ACE 2004 dataset. We report our experimental results in Sec. 6 and conclude with discussion in Sec. 7.

We conclude the introduction by giving a high-level overview of our system in Fig. 2.

Input: document d; mentions $M = \{m_1, \ldots, m_N\}$
1) For each $m_i \in M$, assign it a Wikipedia page p_i in a context-sensitive way (p_i may be $null$).
- If $p_i \neq null$: extract knowledge attributes from p_i and assign to m.
- Else extract knowledge attributes directly from m via noun-phrase parsing techniques (Vadas and Curran, 2008).
3) Let $Q = \{(m_i, m_j)\}_{i \neq j}$, be the queue of mention pairs *approximately* sorted by "easy-first" (Goldberg and Elhadad, 2010).
4) Let G be a partial clustering graph.
5) While Q is not empty
- Extract a pair $p = (m_i, m_j)$ from Q.
- Using the knowledge attributes of m_i and m_j as well as the structure of G, classify whether p is co-referent.
- Update G with the classification decision.
6) Construct an end clustering from G.

Figure 2: High-level system architecture.

2 Baseline System

In this work, we are using the state-of-the-art system of (Bengtson and Roth, 2008), which relies on a pairwise scoring function pc to assign an ordered pair of mentions a probability that they are coreferential. It uses a rich set of features including: string edit distance, gender match, whether the mentions appear in the same sentence, whether the heads are synonyms in WordNet etc. The function pc is modeled using regularized averaged perceptron for a tuned number of training rounds, learning rate and margin. For the end system, we keep these parameters intact, our only modifications will be adding knowledge-rich features and adding intermediate classification sieves to the training and the inference, which we will discuss in the following sections.

At inference time, given a document d and a pairwise co-reference scoring function pc, (Bengtson and Roth, 2008) generate a graph G_d accord-

[2]The extracted attributes and the related resources are available for public download at `http://cogcomp.cs.illinois.edu/Data/Ace2004CorefWikiAttributes.zip`

ing to the Best-Link decision model (Ng and Cardie, 2002) as follows. For each mention m in document d, let B_m be the set of mentions appearing before m in d. Let a be the highest scoring antecedent: $a = \text{argmax}_{b \in B_m}(pc(b, m))$. We will add the edge (a, m) to G_d if $pc(a, m)$ predicts the pair to be co-referent with a confidence exceeding a chosen threshold, then we take the transitive closure[3].

The properties of the Best-Link inference are illustrated in Fig. 3. At this stage, we ask the reader to ignore the knowledge attributes at the bottom of the figure. Let us assume that the pairwise classifier labeled the mentions (m_2, m_5) co-referent because they have identical surface form; mentions (m_1, m_4) are co-referred because the heads are synonyms in WordNet. Let us assume that since m_1 and m_2 appear in the same sentence, the pairwise classifier managed to leverage the dependency parse tree to correctly co-ref the pair (m_1, m_2). The transitive closure would correctly link (m_1, m_5) despite the incorrect prediction of the pairwise classifier on (m_1, m_5), and would incorrectly link m_4 with all other mentions because of the incorrect pairwise prediction on (m_1, m_4) and despite the correct predictions on (m_2, m_4) and (m_4, m_5).

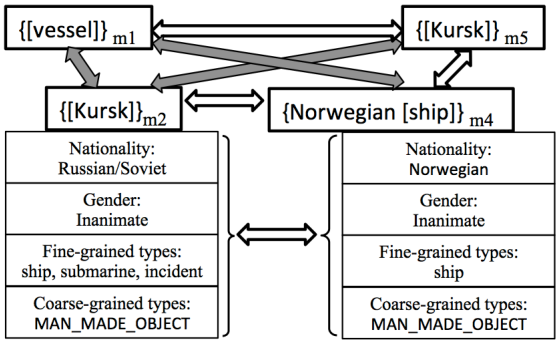

Figure 3: A sample output of a pairwise co-reference classifier. The full edges represent a co-ref prediction and the empty edges represent a non-coref prediction. A set of knowledge attributes for selected mentions is shown as well.

3 Wikipedia as Knowledge

In this section we describe our methodology for using Wikipedia as a knowledge resource. In Sec. 3.1 we cover the process of knowledge extraction from Wikipedia pages. We describe how to inject this knowledge into mentions in Sec. 3.2. The bottom part of Fig. 3 illustrates the knowledge attributes our system injects to two sample mentions at this stage. Finally, in Sec. 3.3 we describe a compatibility metric our system learns over the injected knowledge.

3.1 Wikipedia Knowledge Attributes

Our goal in this section is to extract from Wikipedia pages a compact and highly-accurate set of **knowledge attributes**, which nevertheless possesses discriminative power for co-reference[4]. We concentrate on three types of knowledge attributes: fine-grained semantic categories, gender information and nationality where applicable.

Each Wikipedia page is assigned a set of categories. There are over 100K categories in Wikipedia, many are extremely fine-grained and contain very few pages. The value of the Wikipedia category structure for knowledge acquisition has long been noticed in several influential works, such as (Suchanek et al., 2007; Nastase and Strube, 2008) to name a few. However, while the recall of the above resources is excellent, we found their precision insufficient for our purposes. We have implemented a simple high-precision low-recall heuristic for extracting the head words of Wikipedia categories as follows.

We noticed that Wikipedia categories have a simple structure of either *<noun-phrase>* or *<noun-phrase><relation-token><noun-phrase>*, where in the second case the category information is always on the left. Therefore, we first remove the text succeeding a set of carefully chosen relation tokens[5]. With this heuristic *"Recipients of the Gold Medal of the Royal Astronomical Society"* becomes just *"Recipients"*; *"Populated places in Africa"* becomes *"places"*; however *"Institute for Advanced Study faculty"* becomes *"Institute"* (rather than *"faculty"*). At the second step, we apply the Illinois POS tagger and keep only the tokens labeled as NNS. This step allows us to exclude singular nouns incorrectly identified as heads, such as *"Institute"* above. To further reduce the noise in the category

[3] We use Platt Scaling while (Bengtson and Roth, 2008) used the raw output value of *pc*.

[4] We justify the reasons for our choice of high-precision low-recall knowledge extraction in Sec. 3.2.

[5] The selected set was: {of, in, with, from, ",", at, who, which, for, and, by}

extraction, we also remove all rare category tokens which appeared in less than 100 titles ending up with 2088 fine-grained entity types. We manually map popular fine-grained categories to coarser-grained ones, more consistent with ACE entity typing. A sample of the mapping is shown in the table below:

Fine-grained	Coarse-grained
departments, organizations, banks, ...	ORG
venues, trails, areas, buildings, ...	LOC
countries, towns, villages, ...	GPE
churches, highways, schools, ...	FACILITY

Manual inspection of the extracted category keywords has led us to believe that this heuristic achieves a higher precision at a considerable loss of recall when compared to the more sophisticated approach of (Nastase and Strube, 2008), which correctly identifies *"faculty"* as the head of *"Institute for Advanced Study faculty"*, but incorrectly identifies *"statistical organizations"* as the head of *"Presidents of statistical organizations"* in about half the titles containing the category[6].

We assign gender to the titles using the following simple heuristic. The first paragraph of each Wikipedia article provides a very brief summary of the entity in focus. If the first paragraph of a Wikipedia page contains the pronoun "she", but not "he", the article is considered to be about a female (and vice-versa). However, when the page is assigned a non-person-related fine-grained NE type (e.g. school) and at the same time is not assigned a person-related fine-grained NE type (e.g. novelist), we mark the page as inanimate regardless of the presence of he/she pronouns. The nationality is assigned by matching the tokens in the original (unprocessed) categories of the Wikipedia page to a list of countries. We assign nationality not only to the Wikipedia titles, but also to single tokens. For each token, we track the list of titles it appears in, and if the union of the nationalities assigned to the titles it appears in is less than 7, we mark the token compatible with these nationalities. This allows us to identify Ivan Lebedev as Russian and Ronen Ben-Zohar as Israeli, even though Wikipedia may not contain pages for these specific people.

[6] (Nastase and Strube, 2008) analyze a set of categories S assigned to Wikipedia page p jointly, hence the same category expression can be interpreted differently, depending on S.

3.2 Injecting Knowledge Attributes

Once we have extracted the knowledge attributes of Wikipedia pages, we need to inject them into the mentions. (Rahman and Ng, 2011) used YAGO for similar purposes, but noticed that knowledge injection is often noisy. Therefore they used YAGO only for mention pairs where one mention was an NE of type PER/LOC/ORG and the other was a common noun. This implies that all MISC NEs were discarded, and all NE-NE pairs were discarded as well. We also note that (Rahman and Ng, 2011) reports low utility of FrameNet-based features. In fact, when incrementally added to other features in cluster-ranking model the FrameNet-based features sometimes led to performance drops. This observation has motivated our choice of high-precision low-recall heuristic in Sec. 3.1 and will motivate us to add features conservatively when building attribute compatibility metric in Sec. 3.3.

Additionally, while (Rahman and Ng, 2011) uses the union of all possible meanings a mention may have in Wikipedia, we deploy GLOW (Ratinov et al., 2011)[7], a context-sensitive system for disambiguation to Wikipedia. Using context-sensitive disambiguation to Wikipedia as well as high-precision set of knowledge attributes allows us to inject the knowledge to more mention pairs when compared to (Rahman and Ng, 2011). Our exact heuristic for injecting knowledge attributes to mentions is as follows:

Named Entities with Wikipedia Disambiguation
If the mention head is an NE matched to a Wikipedia page p by GLOW, we import all the knowledge attributes from p. GLOW allows us to map *"E***ph***raim Sneh"* to *http://en.wikipedia.org/wiki/E***f***raim_Sneh* and to assign it the *Israeli* nationality, *male* gender, and the fine-grained entity types: {*member, politician, person, minister, alumnus, physician, general*}.

Head and Extent Keywords
If the mention head is not mapped to Wikipedia by GLOW and the head contains keywords which appear in the list of 2088 fine-grained entity types, then the rightmost such keyword is added to the list of mention knowledge attributes. If the head does

[7] Available at: `http://cogcomp.cs.illinois.edu/page/software_view/Wikifier`

not contain any entity-type keywords but the extent does, we add the rightmost such keyword of the extent. In both cases, we apply the heuristic of removing clauses starting with a select set of punctuations, prepositions and pronouns, annotating what is left with POS tagger and restricting to noun tokens only[8]. This allows us to inject knowledge to mentions unmapped to Wikipedia, such as: *"{current Cycle World publisher [Larry Little]}"*, which is assigned the attribute *publisher* but not *world* or *cycle*. Likewise, *"{[Joseph Conrad Parkhurst], who founded the motorcycle magazine Cycle World in 1962 }"*, is not assigned the attribute *magazine*, since the text following *"who"* is discarded.

3.3 Learning Attributes Compatibility

In the previous section we have assigned knowledge attributes to the mentions. Some of this information, such as gender and coarse-grained entity types are also modeled in the baseline system of (Bengtson and Roth, 2008). Our goal is to build a compatibility metric on top of this redundant, yet often inconsistent information.

The majority of the features we are using are straightforward, such as: (1) whether the two mentions mapped to the same Wikipedia page, (2) gender agreement (both Wikipedia and dictionary-based), (3) nationality agreement (here we measure only whether the sets intersect, since mentions can have multiple nationalities in the real world), (4) coarse-grained entity type match, etc.

The only non-trivial feature is measuring compatibility between sets of fine-grained entity types, which we describe below. Let us assume that mention m_1 was assigned the set of fine-grained entity types S_1 and the mention m_2 was assigned the set of fine-grained entity types S_2. We record whether S_1 and S_2 share elements. If they do, than, in addition to the Boolean feature, the list of the shared elements also appears as a list of discrete features. We do the same for the most similar and most dissimilar elements of S_1 and S_2 (along with their discretized similarity score) according to a WordNet-based similarity metric of (Do et al., 2009). The reason for explicitly listing the shared, the most similar and dissimilar elements is that the WordNet similarity does not always correspond to co-reference compatibility. For example, the pair $(company, rival)$ has a low similarity score according to WordNet, but characterizes co-referent mentions. On the other hand, the pair $(city, region)$ has a high WordNet similarity score, but characterizes non-coreferent mentions. We want to allow our system to "memorize" the discrepancy between the WordNet similarity and co-reference compatibility of specific pairs.

We also note that we generate a set of selected conjunctive features, most notably of fine-grained categories with NER predictions. The reason is that the pair of mentions *"(Microsoft, Google)"* are not co-referent despite the fact that they both have the *company* attribute. On the other hand *"(Microsoft, Redmond-based company)"* is a co-referent pair. To capture this difference, we generate the feature *ORG-ORG&&share_attribute* for the first pair, and *ORG-O&&share_attribute* for the second pair[9]. These features are also used in conjunction with string edit distance. Therefore, if our system sees two named entities which share the same fine-grained type but have a large string edit distance, it will label the pair as non-coref.

4 Learning-based Multi-Sieve Aproach

State-of-the-art machine-learning co-ref systems, e.g. (Bengtson and Roth, 2008; Rahman and Ng, 2011) train a single model for predicting co-reference of all mention pairs. However, rule-based systems, e.g. (Haghighi and Klein, 2009; Raghunathan et al., 2010) characterize mention pairs by discourse structure and linguistic properties and apply rules in a prescribed order (high-precision rules first). Somewhat surprisingly, such hybrid approach of applying rules on top of structures produced by statistical tools (such as dependency parse trees) performs better than pure machine-learning approach[10].

In this work, we attempt to integrate the strength of linguistically motivated rule-based systems with the robustness of a machine learning approach. We started with a hypothesis that different types of men-

[8]This heuristic is similar to the one we used for extracting Wikipedia category headwords and seems to be a reasonable baseline for parsing noun structures (Vadas and Curran, 2008).

[9]The head of *"Redmond-based company"* is *"company"*, which is not a named entity, and is marked O.

[10](Raghunathan et al., 2010) recorded the best result on CoNLL 2011 shared task.

tion pairs may require a different co-ref model. For example, consider the text below:

Queen Rania of Jordan , Egypt's [*Suzanne Mubarak*]$_{m_1}$ *and others were using their charisma and influence to campaign for equality of the sexes.* [*Mubarak*]$_{m_2}$*, wife of Egyptian President* [*Hosni Mubarak*]$_{m_3}$*, and one of the conference organizers, said they must find ways to ...*

There is a subtle difference between mention pairs (m_1, m_2) and (m_2, m_3). One of the differences is purely structural. The first pair appears in different sentences, while the second pair – in the same sentence. It turns out that string edit distance feature between two named entities has different "semantics" depending on whether the two mentions appear in the same sentence. The reason is that to avoid redundancy, humans refer to the same entity differently within the sentence, preferring titles, nicknames and pronouns. Therefore, when a similar-looking named entities appear in the same sentence, they are actually likely to refer to different entities. On the other hand, in the sentence *"Reggie Jackson, nicknamed Mr. October ..."* we have to rely heavily on sentence structure rather than string edit distance to make the correct co-ref prediction.

Sieve	Trained on All Data	Sieve-specific Training
AllSentencePairs	61.37	67.46
ClosestNonProDiffSent	60.71	63.33
NonProSameSentence	62.97	63.80
NerMentionsDiffSent	86.44	87.12
SameSentenceOneNer	64.10	68.88
Adjacent	71.00	78.80
SameSenBothNer	75.30	73.75
Nested	76.11	79.00

Table 1: F1 performance on co-referent mention pairs by sieve type when trained with all data versus sieve-specific data only.

Our second intuition is that "easy-first" inference is necessary to effectively leverage knowledge. For example, in Fig. 3, our goal is to link *vessel* to *Kursk* and assign it the *Russian/Soviet* nationality prior to applying the pairwise co-reference classifier on (*vessel*, *Norwegian ship*). Therefore, our goal is to apply the pairwise classifier on pairs in *prescribed order* and to propagate the knowledge across mentions. The ordering should be such that (a) maximum amount of information is injected at early stages (b) the precision at the early stages is as high as possible (Raghunathan et al., 2010). Hence, we divide the mention pairs as follows:

Nested: are pairs such as *"*{{[*city*]$_{m_1}$} *of* [*Jerusalem*]$_{m_2}$}*"* where the extent of one of the mentions contains the extent of the other. For some mentions, the extent is the entire clause, so we also added a requirement that mention heads are at most 7 tokens apart. Intuitively, it is the easiest case of co-reference. There are 5,804 training samples and 992 testing samples, out of which 208 are co-referent.

SameSenBothNer: are pairs of named entities which appear in the same sentence. We already saw an example for this case involving [Mubarak]$_{m2}$ and [Hosni Mubarak]$_{m3}$. There are 13,041 training samples and 1,746 testing samples, out of which 86 are co-referent.

Adjacent: are pairs of mentions which appear closest to each other on the dependency tree. We note that most of the nested pairs are also adjacent. There are training 5,872 samples and 895 testing samples, out of which 219 are co-referent.

SameSentenceOneNer: are pairs which appear in the same sentence and exactly one of the mentions is a named entity, and the other is not a pronoun. Typical pairs are *"Israel-country"*, as opposed to *"Bill Clinton - reporter"*. This type of pairs is fairly difficult, but our hope is to use encyclopedic knowledge to boost the performance. There are 15,715 training samples and 2,635 testing samples, out of which 207 are co-referent.

NerMentionsDiffSent: are pairs of mentions in different sentences, both of which are named entities. There are 189,807 training samples and 24,342 testing samples, out of which 1,628 are co-referent.

NonProSameSentence: are pairs in the same sentence, where both mentions are non-pronouns. This sieve includes all the pairs in the SameSentenceOneNer sieve. Typical pairs are *"city-capital"* and *"reporter-celebrity"*. *There are 33,895 training samples and 5,393 testing samples, out of which 336 are co-referent.*.

ClosestNonProDiffSent: are pairs of mentions in different sentences with no other mentions between the two. 3,707 training samples and 488 testing samples, out of which 38 are co-referent.

AllSentencePairs: All mention pairs within same sentence. There are 49,953 training samples and 7,809 testing samples, out of which 846 are co-referent.

TopSieve: The set of mention pairs classified by the baseline system. 525,398 training samples and 85,358 testing samples, out of which 1,387 are co-referent.

In Tab. 1 we compare the performance at each sieve in two scenarios[11]. First, we train with the entire 525,398 training samples, and then we train on

[11]The data is described in Sec. 6.1.

whatever training data is available for the specific sieve[12]. We were surprised to see that the F1 on the nested mentions, when trained on the 5,804 sieve-specific samples improves to 79.00 versus 76.11 when trained on the 525,398 top sieve samples.

There are several things to note when interpreting the results in Tab 1. First, the sheer ratio of positive to negative samples fluctuates drastically. For example, 208 out of the 992 testing samples at the nested sieve are positive, while there are only 86 positive samples out of 1,746 testing samples in the SameSenBothNer sieve. It seems unreasonable to use the same model for inference at both sieves. Second, the data for intermediate sieves is not always a subset of the top sieve. The reason is that top sieve extracts a positive instance only for the closest co-referent mentions, while sieves such as AllSentencePairs extract samples for all co-referent pairs which appear in the same sentence. Third, while our division to sieves may resemble witchcraft, it is motivated by the intuition that mentions appearing close to one another are easier instances of co-ref as well as linguistic insights of (Raghunathan et al., 2010).

5 Entity-Based Features

In this section we describe our approach for building entity-based features. Let $\{C_1, C_2, \ldots C_N\}$ be the set of sieve-specific classifiers. In our case, C_1 is the nested mention pairs classifier, C_2 is the SameSenBothNer classifier, and C_9 is the top sieve classifier. We design entity-based features so that the subsequent sieves "see" the decisions of the previous sieves and use entity-based features based on the intermediate clustering. However, unlike (Raghunathan et al., 2010), we allow the subsequent sieves to change the decisions made by the lower sieves (since additional information becomes available).

5.1 Intermediate Clustering Features (IC)

Let $R_i(m)$ be the set of all mentions which, when paired with the mention m, form valid sample pairs for sieve i. E.g. in our running example of Fig. 1, $R_2([Kursk]_{m_2}) = \{[\textit{Barents Sea}]_{m_3}\}$, since both m_1 and m_2 are NEs and appear in the same sentence. Let $R_i^+(m)$ be the set of all mentions which were labeled as co-referent to the mention m by the classifier C_i (including m, which is co-referent to itself). We define $R_i^-(m)$ similarly. We denote the union of mentions co-refed to m during inference up to sieve i as $E_i^+(m) = \cup_{j=1}^{i-1} R_j^+(m)$. Similarly, $E_i^-(m) = \cup_{j=1}^{i-1} R_j^-(m)$. Using these definitions we can introduce entity-based prediction features which allow subsequent sieves to use information aggregated from previous sieves:

$$IC_i^R(m_j, m_k) = \begin{cases} -1 & m_j \in R_{i-1}^-(m_k) \\ +1 & m_j \in R_{i-1}^+(m_k) \\ 0 & \text{Otherwise} \end{cases}$$

$$IC_i^E(m_j, m_k) = \begin{cases} -1 & m_j \in E_{i-1}^-(m_k) \\ +1 & m_j \in E_{i-1}^+(m_k) \\ 0 & \text{Otherwise} \end{cases}$$

IC_i^R stores the pairwise prediction history, thus when classifying a pair (m_j, m_k) at sieve i, a classifier can see the predictions of all the previous sieves applicable on that pair. IC_i^E stores the transitive closures of the sieve-specific predictions. We note that both IC_i^R and IC_i^E can have the values +1 and -1 active at the same time if intermediate sieve classifiers generated conflicting predictions. However, a classifier at sieve i will use as features both $IC_1^R, \ldots IC_{i-1}^R$ and $IC_1^E, \ldots IC_{i-1}^E$, thus it will know the lowest sieve at which the conflicting evidence occurs. The classifier at sieve i also uses set identity, set containment, set overlap and other set comparison features between $E_{i-1}^{+/-}(m_j)$ and $E_{i-1}^{+/-}(m_k)$. We check whether the sets have symmetric difference, whether the size of the intersection between the two sets is at least half the size of the smallest set etc. We also generate subtypes of set comparison features when restricting the elements to NE-mentions and non-pronominal mentions (e.g "what percent of named entities do the sets have in common?").

5.2 Surface Form Compatibility (SFC)

The intermediate clustering features do not allow us to generalize predictions from pairs of mentions to pairs of surface strings. For example, if we have three mentions: $\{[\textit{vessel}]_{m_1}, [\textit{Kursk}]_{m_2}, [\textit{Kursk}]_{m_5}\}$, then the prediction on the pair (m_1, m_2) will not be

[12] We report *pairwise* performance on mention pairs because it is the more natural metric for the intermediate sieves. We report only performance on co-referent pairs, because for many sieves, such as the top sieve, 99% of the mention pairs are non-coreferent, hence the baseline of labeling all samples as non-coreferent would result in 99% accuracy. We are interested in a more challenging baseline, the co-referent pairs.

	(B)aseline	(B)+Knowledge	(B)+Predictions	(B)+Knowledge+Predictions
TopSieve	66.58	69.08	68.77	**70.43**
AllSentencePairs	67.46	71.79	69.59	**73.50**
ClosestNonProDiffSent	63.33	65.62	65.57	**70.76**
NonProSameSentence	63.80	69.62	67.03	**71.11**
NerMentionsDiffSent	87.12	88.23	88.68	**89.07**
SameSentenceOneNer	68.88	70.58	67.89	**73.17**
Adjacent	78.80	81.32	80.00	**81.79**
SameSenBothNer	73.75	80.50	77.21	**80.98**
Nested	79.00	83.59	80.65	**83.37**

Table 2: Utility of knowledge and prediction features (F1 on co-referent mention pairs) by inference sieves. Both knowledge and entity-based features significantly and independently improve the performance for all sieves. The goal of entity-based features is to propagate knowledge effectively, thus it is encouraging that the combination of entity-based and knowledge features performed significantly better than any of the approaches individually at the top sieve.

used for the prediction on the pair (m_1, m_5), even though in both pairs we are asking whether *Kursk* can be referred to as *vessel*. The surface form compatibility features mirror the intermediate clustering features, but relax mention IDs and replace them by surface forms. Similarly to intermediate clustering features, both +1 and -1 values can be active at the same time. We also generate subtypes of set-comparison features for NE-mentions and optionally stemmed non-pronominal mentions. For example, in a text discussing *President Clinton* and *President Putin*, some instances of the surface from *president* will refer to *Putin* but not *Clinton* and vice-versa. Therefore, both for *(Putin, president)* and for *(Clinton, president)*, the surface from compatibility will be +1 and -1 simultaneously. This indicates to the system that *Putin* can be referred to as *president*, but *president* can refer to other entities in the document as well.

6 Experiments and Results

6.1 Data

We use the official ACE 2004 English training data (NIST, 2004). We started with the data split used in (Culotta et al., 2007), which used 336 documents for training and 107 documents for testing. We note that ACE data contains both newswire text and transcripts. In this work, we are using NLP tools such as POS tagger, named entity recognizer, shallow parser, and a disambiguation to Wikipedia system to inject expressive features into a co-reference system.

Unfortunately, current state-of-the-art NLP tools do not work well on transcribed text. Therefore, we discard all the transcripts. Our criteria was simple. The ACE annotators have marked the named entities both in newswire and in the transcripts. We kept only those documents which contained named entities (according to manual ACE annotation) and at least $1/3$ of the named entities started with a capital letter. After this pre-processing step, we were left with 275 out of the original 336 training documents, and 42 out of the 107 testing documents.

For the experiments throughout this paper, following Culotta et al. (Culotta et al., 2007) and much other work, to make experiments more comparable across systems, we assume that perfect mention boundaries and mention type labels are given. However, we do not use the gold named entity types such as person/location/facility etc. available in the data. In all experiments we automatically split words and sentences, and annotate the text with part-of-speech tags, named entities and cross-link concepts from the text to Wikipedia using publicly available tools.

6.2 Ablation Study

In Tab. 2 we report the pairwise F1 scores on co-referent mention pairs broken down by sieve and using different components. This allows us to see, for example, that adding only the knowledge attributes improved the performance at *NonProSameSentence* sieve from 63.80 to 69.62. We have ordered the sieves according to our initial intuition of "easy first". We were surprised to see that co-ref resolution for named entities in the same sentence was harder than cross-sentence (73.75 vs. 87.12 base-

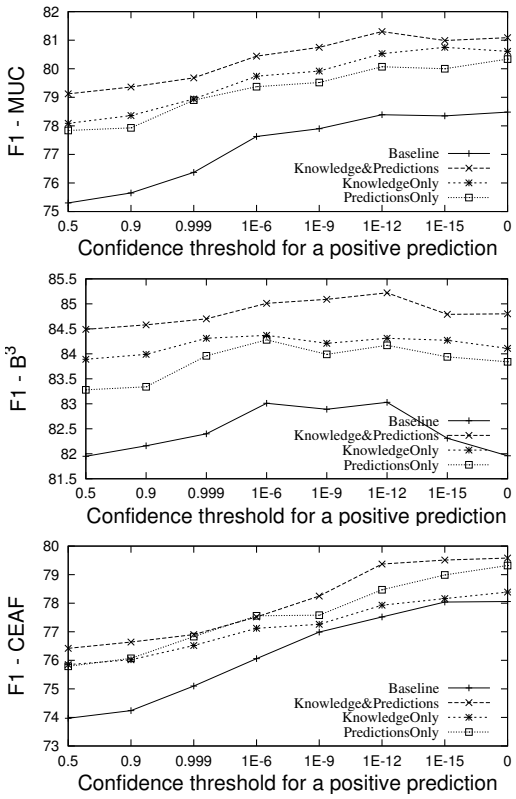

Figure 4: End performance for various systems.

line F1). We were also surprised to see that resolving all mention pairs within sentence when including pronouns was easier than resolving pairs where both mentions were non-pronouns (67.46 vs. 63.80 baseline F1).

We note that conceptually, the nested (B)+Predictions sieve should be identical to the baseline. However, in practice, the surface form compatibility (SFC) features are generated for the nested sieve as well. Given two mentions m_1 and m_2, the SFC features capture how many surface forms $E^+(m_1)$ and $E^+(m_2)$ share. At the nested sieve, $E^+(m)$ and $R^+(m)$ are just m, which is identical to string comparison features already existing in the baseline system. While the SFC features do not add new information, they influence the weight the features get (essentially leading to a different regularization), which in turn leads to slightly different results.

6.3 End system performance

Recall that the Best-Link algorithm applies transitive closure on the graph generated by thresholding the pairwise co-reference scoring function pc. The lower the threshold on the positive prediction, the lower is the precision and the higher is the recall. In Fig. 4 we compare the end clustering quality across a variety of thresholds and for various system flavors using three metrics: MUC (Vilain et al., 1995), B^3 (Bagga and Baldwin, 1998) and CEAF (Luo, 2005)[13]. The purpose of this comparison is to see the impact of the knowledge and the prediction features on the final output and to see whether the performance gains are due to (mis-)tuning of one of the systems or are they consistent across a variety of thresholds.

The end performance of the baseline system on our training/testing split peaks at around 78.39 MUC, 83.03 B^3 and 77.52 CEAF, which is higher (e.g. 3 B^3 F1 points) than the originally reported result on the entire dataset (which includes the transcripts). This is expected, since well-formed text is easier to process than transcripts. We note that our baseline is a state-of-the art system which recorded the highest B^3 and BLANC scores at CoNLL 2011 shared task and took the third place overall. Fig. 4 shows a minimum improvement of 3 MUC, 2 B^3 and 1.25 CEAF F1 points across all thresholds when comparing the baseline to our end system. Surprisingly, the knowledge features outperformed prediction features on pairwise, MUC and B^3 metrics, but not on the CEAF metric. This shows that pairwise performance is not always indicative of cluster-level performance for all metrics.

7 Conclusions and Related Work

To illustrate the strengths of our approach, let us consider the following text:

Another terminal was made available in $\{[Jiangxi]_{m_1}\}$*, an* $\{$*inland* $[province]_{m_2}\}$*. ... The previous situation whereby large amount of goods for* $\{$*Jiangxi* $[province]_{m_3}\}$ *had to be re-shipped through Guangzhou and Shanghai will be changed completely.*

The baseline system assigns each mention to a separate cluster. The pairs (m_1, m_2) and (m_1, m_3)

[13] In the interest of space, we refer the reader to the literature for details about the different metrics.

are misclassified because the baseline classifier does not know that *Jiangxi* is a *province* and the preposition *an* before m_2 is interpreted to mean it is a previously unmentioned entity. The pair (m_2, m_3) is misclassified because identical heads have different modifiers, as in *(big province, small province)*. Our end system first co-refs (m_1, m_2) at the *AllSameSentence* sieve due to the knowledge features, and then co-refs (m_1, m_3) at the top sieve due to surface form compatibility features indicating that *province* was observed to refer to *Jiangxi* in the document. The transitivity of Best-Link takes care of (m_2, m_3).

However, our approach has multiple limitations. Entity-based features currently do not propagate knowledge attributes directly, but through aggregating pairwise predictions at knowledge-infused intermediate sieves. We rely on gold mention boundaries and exhaustive gold co-reference annotation. This prevented us from applying our approach to the Ontonotes dataset where singleton clusters and co-referent nested mentions are removed. Therefore the gold annotation for training several sieves of our scheme is missing (e.g. nested mentions). Another limitation is our somewhat preliminary division to sieves. (Vilalta and Rish, 2003) have experimented with approaches for automatic decomposition of data to subclasses and learning multiple models to improve data separability. We hope that similar approach would be useful for co-reference resolution. Ideally, we want to make "simple decisions" first, similarly to what was done in (Goldberg and Elhadad, 2010) for dependency parsing, and model clustering as a structured problem, similarly to (Joachims et al., 2009; Wick et al., 2011). However, our experience with multi-sieve approach with classifiers suggests that a single model would not perform well for both lower sieves with little entity-based information and higher sieves with a lot of entity-based features. Addressing the aforementioned challenges is a subject for future work.

There has been an increasing interest in knowledge-rich co-reference resolution (Ponzetto and Strube, 2006; Haghighi and Klein, 2010; Rahman and Ng, 2011). We use Wikipedia differently from (Ponzetto and Strube, 2006) who focus on using WikiRelate, a Wikipedia-based relatedness metric (Strube and Ponzetto, 2006). (Rahman and Ng, 2011) used the union of all possible interpretations a mention may have in YAGO, which means that *Michael Jordan* could be co-refed both to a *scientist* and *basketball player* in the same document. Additionally, (Rahman and Ng, 2011) use exact word matching, relying on YAGO's ability to extract a comprehensive set of facts offline[14]. We are the first to use context-sensitive disambiguation to Wikipedia, which received a lot of attention recently (Bunescu and Pasca, 2006; Cucerzan, 2007; Mihalcea and Csomai, 2007; Milne and Witten, 2008; Ratinov et al., 2011). We extract context-sensitive, high-precision knowledge attributes from Wikipedia pages and apply (among other features) WordNet similarity metric on pairs of knowledge attributes to determine attribute compatibility.

We have integrated the strengths of rule-based systems such as (Haghighi and Klein, 2009; Raghunathan et al., 2010) into a multi-sieve machine learning framework. We show that training sieve-specific models significantly increases the performance on most intermediate sievesieves.

We develop a novel approach for entity-based inference. Unlike (Rahman and Ng, 2011) who construct entities left-to-right, and similarly to (Raghunathan et al., 2010) we resolve easy instances of coref to reduce error propagation in entity-based features. Unlike (Raghunathan et al., 2010), we allow later stages of inference to change the decisions made at lower stages as additional entity-based evidence becomes available.

By adding word-knowledge features and refining the inference, we improve the performance of a state-of-the-art system of (Bengtson and Roth, 2008) by 3 MUC, 2 B^3 and 2 CEAF F1 points on the non-transcript portion of the ACE 2004 dataset.

References

A. Bagga and B. Baldwin. 1998. Algorithms for scoring coreference chains. In *MUC7*.

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

[14] YAGO uses WordNet to expand its set of facts. For example, if *Martha Stewart* is assigned the meaning *personality* from category head words analysis, YAGO adds the meaning *celebrity* because *personality* is a direct hyponym of *celebrity* in WordNet. However, this is done offline in a context-insensitive way, which is inherently limited.

R. C. Bunescu and M. Pasca. 2006. Using encyclopedic knowledge for named entity disambiguation. In *EACL*.

S. Cucerzan. 2007. Large-scale named entity disambiguation based on Wikipedia data. In *EMNLP-CoNLL*.

A. Culotta, M. Wick, R. Hall, and A. McCallum. 2007. First-order probabilistic models for coreference resolution. In *HLT/NAACL*, pages 81–88.

Q. Do, D. Roth, M. Sammons, Y. Tu, and V. Vydiswaran. 2009. Robust, light-weight approaches to compute lexical similarity. Technical report, University of Illinois at Urbana-Champaign.

A. Fader, S. Soderland, and O. Etzioni. 2009. Scaling wikipedia-based named entity disambiguation to arbitrary web text. In *WikiAI (IJCAI workshop)*.

Y. Goldberg and M. Elhadad. 2010. An efficient algorithm for easy-first non-directional dependency parsing. In *NAACL*.

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-ACL*. Association for Computational Linguistics.

T. Joachims, T. Hofmann, Y. Yue, and C. Yu. 2009. Predicting structured objects with support vector machines. *Communications of the ACM, Research Highlight*, 52(11):97–104, November.

X. Luo. 2005. On coreference resolution performance metrics. In *HLT*.

R. Mihalcea and A. Csomai. 2007. Wikify!: linking documents to encyclopedic knowledge. In *CIKM*.

D. Milne and I. H. Witten. 2008. Learning to link with wikipedia. In *CIKM*.

V. Nastase and M. Strube. 2008. Decoding wikipedia categories for knowledge acquisition. In *AAAI*.

V. Ng and C. Cardie. 2002. Improving machine learning approaches to coreference resolution. In *ACL*.

NIST. 2004. The ace evaluation plan. www.nist.gov/speech/tests/ace/index.htm.

S. P. Ponzetto and M. Strube. 2006. Exploiting semantic role labeling, wordnet and wikipedia for coreference resolution. In *HLT-ACL*.

K. Raghunathan, H. Lee, S. Rangarajan, N. Chambers, M. Surdeanu, D. Jurafsky, and C. Manning. 2010. A multi-pass sieve for coreference resolution. In *EMNLP*.

A. Rahman and V. Ng. 2011. Coreference resolution with world knowledge. In *HLT-ACL*.

L. Ratinov, D. Downey, M. Anderson, and D. Roth. 2011. Local and global algorithms for disambiguation to wikipedia. In *ACL*.

M. Strube and S. P. Ponzetto. 2006. WikiRelate! Computing Semantic Relatedness Using Wikipedia. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, July.

F. M. Suchanek, G. Kasneci, and G. Weikum. 2007. Yago: A core of semantic knowledge. In *WWW*.

D. Vadas and J. R. Curran. 2008. Parsing noun phrase structure with CCG. In *ACL*.

M. Vilain, J. Burger, J. Aberdeen, D. Connolly, and L. Hirschman. 1995. A model-theoretic coreference scoring scheme. In *MUC6*, pages 45–52.

R. Vilalta and I. Rish. 2003. A decomposition of classes via clustering to explain and improve naive bayes. In *ECML*.

M. Wick, K. Rohanimanesh, K. Bellare, A. Culotta, and A. McCallum. 2011. Samplerank: Training factor graphs with atomic gradients. In *ICML*.