Same Referent, Different Words: Unsupervised Mining of Opaque Coreferent Mentions

Marta Recasens*, **Matthew Can**†, **and Dan Jurafsky***
*Linguistics Department, Stanford University, Stanford, CA 94305
†Computer Science Department, Stanford University, Stanford, CA 94305
recasens@google.com, {mattcan,jurafsky}@stanford.edu

Abstract

Coreference resolution systems rely heavily on string overlap (e.g., *Google Inc.* and *Google*), performing badly on mentions with very different words (**opaque** mentions) like *Google* and *the search giant*. Yet prior attempts to resolve opaque pairs using ontologies or distributional semantics hurt precision more than improved recall. We present a new unsupervised method for mining opaque pairs. Our intuition is to *restrict* distributional semantics to articles about the same event, thus promoting referential match. Using an English comparable corpus of tech news, we built a dictionary of opaque coreferent mentions (only 3% are in WordNet). Our dictionary can be integrated into any coreference system (it increases the performance of a state-of-the-art system by 1% F1 on all measures) and is easily extendable by using news aggregators.

1 Introduction

Repetition is one of the most common coreferential devices in written text, making string-match features important to all coreference resolution systems. In fact, the scores achieved by just head match and a rudimentary form of pronominal resolution[1] are not far from that of state-of-the-art systems (Recasens and Hovy, 2010). This suggests that **opaque** mentions (i.e., lexically different) such as *iPad* and *the Cupertino slate* are a serious problem for modern systems: they comprise 65% of the non-pronominal errors made by the Stanford system on the CoNLL-2011 data. Solving this problem is critical for overcoming the recall gap of state-of-the-art systems (Haghighi and Klein, 2010; Stoyanov et al., 2009).

Previous systems have turned either to ontologies (Ponzetto and Strube, 2006; Uryupina et al., 2011; Rahman and Ng, 2011) or distributional semantics (Yang and Su, 2007; Kobdani et al., 2011; Bansal and Klein, 2012) to help solve these errors. But neither semantic similarity nor hypernymy are the same as coreference: *Microsoft* and *Google* are distributionally similar but not coreferent; *people* is a hypernym of both *voters* and *scientists*, but *the people* can corefer with *the voters*, but is less likely to corefer with *the scientists*. Thus ontologies lead to precision problems, and to recall problems like missing NE descriptions (e.g., *Apple* and *the iPhone maker*) and metonymies (e.g., *agreement* and *wording*), while distributional systems lead to precision problems like coreferring *Microsoft* and *the Mountain View giant* because of their similar vector representation (*release*, *software*, *update*).

We increase precision by drawing on the intuition that referents that are *both* similar *and* participate in the same event are likely to corefer. We restrict distributional similarity to collections of articles that discuss the same event. In the following two documents on the Nexus One from different sources, we take the subjects of the identical verb *release*—*Google* and *the Mountain View giant*—as coreferent.

Document 1: Google **has released** a software update.

Document 2: The Mountain View giant **released** an update.

Based on this idea, we introduce a new unsupervised method that uses verbs in comparable corpora

[1] Closest NP with the same gender and number.

as pivots for extracting the hard cases of coreference resolution, and build a dictionary of opaque coreferent mentions (i.e., the dictionary entries are pairs of mentions). This dictionary is then integrated into the Stanford coreference system (Lee et al., 2011), resulting in an average 1% improvement in the F1 score of all the evaluation measures.

Our work points out the importance of context to decide whether a specific mention pair is coreferent. On the one hand, we need to know what semantic relations are potentially coreferent (e.g., *content* and *video*). On the other, we need to distinguish contexts that are compatible for coreference—(1) and (2-a)—from those that are not—(1) and (2-b).

(1) Elemental helps those big media entities process content across a full slate of mobile devices.

(2) a. Elemental provides the picks and shovels to make video work across multiple devices.
b. Elemental is powering **the video** for HBO Go.

Our dictionary of opaque coreferent pairs is our solution to the first problem, and we report on some preliminary work on context compatibility to address the second problem.

2 Building a Dictionary for Coreference

To build a dictionary of semantic relations that are appropriate for coreference we will use a cluster of documents about the same news event, which we call a **story**. Consider as an example the story *Sprint blocks out vacation days for employees*. We determine using tf-idf the representative verbs for this story, the main actions and events of the story (e.g., *block out*). Since these verbs are representative of the story, different instances across documents in the cluster are likely to refer to the same events (*Sprint blocks out*... and *the carrier blocks out*...). By the same logic, the subjects and objects of the verbs are also likely to be coreferent (*Sprint* and *the carrier*).

2.1 Comparable corpus

To build our dictionary, we require a monolingual **comparable corpus**, containing clusters of documents from different sources that discuss the same story. To ensure likely coreference, the story must be the very same; documents that are merely clustered by (general) topic do not suffice. The corpus does not need to be parallel in the sense that documents in the same cluster do not need to be sentence aligned.

We used Techmeme,[2] a news aggregator for technology news, to construct a comparable corpus. Its website lists the major tech stories, each with links to several articles from different sources. We used the Readability API[3] to download and extract the article text for each document. We scraped two years worth of data from Techmeme and only took stories containing at least 5 documents. Our corpus contains approximately 160 million words, 25k stories, and 375k documents. Using a corpus from Techmeme means that our current coreference dictionary is focused on the technological domain. Our method can be easily extended to other domains, however, since getting comparable corpora is relatively simple from the many similar news aggregator sites.

2.2 Extraction

After building our corpus, we used Stanford's CoreNLP tools[4] to tokenize the text and annotate it with POS tags and named entity types. We parsed the text using the MaltParser 1.7, a linear time dependency parser (Nivre et al., 2004).[5]

We then extracted the representative verbs of each story by ranking the verbs in each story according to their tf-idf scores. We took the top ten to be the representative set. For each of these verbs, we clustered together its subjects and objects (separately) across instances of the verb in the document cluster, excluding pronouns and NPs headed by the same noun. For example, suppose that *crawl* is a representative verb and that in one document we have *Google crawls web pages* and *The search giant crawls sites* in another document. We will create the clusters {*Google*, *the search giant*} and {*web pages*, *sites*}.

When detecting representative verbs, we kept phrasal verbs as a unit (e.g., *give up*) and excluded auxiliary and copular verbs,[6] light verbs,[7] and report

[2] http://www.techmeme.com

[3] http://www.readability.com/developers/api

[4] http://nlp.stanford.edu/software/corenlp.shtml

[5] http://www.maltparser.org

[6] Auxiliary and copular verbs include *appear, be, become, do, have, seem*.

[7] Light verbs include *do, get, give, go, have, keep, make, put, set, take*.

verbs,[8] as they are rarely representative of a story and tend to add noise to our dictionary. To increase recall, we also considered the synonyms from WordNet and nominalizations from NomBank of the representative verbs, thus clustering together the subjects and objects of any synonym as well as the arguments of nominalizations.[9] We used syntactic relations instead of semantic roles because the MaltParser is faster than any SRL system, but we checked for frequent syntactic structures in which the agent and patient are inverted, such as passive and ergative constructions.[10]

From each cluster of subject or object mentions, we generated all pairs of mentions. This forms the initial version of our dictionary. The next sections describe how we filter and generalize these pairs.

2.3 Filtering

We manually analyzed 200 random pairs and classified them into coreference and spurious relations. The spurious relations were caused by errors due to the parser, the text extraction, and violations of our algorithm assumption (i.e., the representative verb does not refer to a unique event). We employed a filtering strategy to improve the precision of the dictionary. We used a total of thirteen simple rules, which are shown in Table 1. For instance, we sometimes get the same verb with non-coreferent arguments, especially in tech news that compare companies or products. In these cases, NEs are often used, and so we can get rid of a large number of errors by automatically removing pairs in which both mentions are NEs (e.g., *Google* and *Samsung*).

Before filtering, 53% of all relations were good coreference relations versus 47% spurious ones. Of the relations that remained after filtering, 74% were coreferent and only 26% were spurious. In total, about half of the dictionary relations were removed in the filtering process, resulting in a total of 128,492 coreferent pairs.

Both mentions are NEs
Both mentions appear in the same document
Object of a negated verb
Enumeration or list environment
Sentence is ill-formed
Number NE
Temporal NE
Quantifying noun
Coordinated
Verb is preceded by a determiner or an adjective
Head is not nominal
Sentence length ≥ 100
Mention length $\geq 70\%$ of sentence length

Table 1: Filters to improve the dictionary precision. Unless otherwise noted, the filter was applied if either mention in the relation satisfied the condition.

2.4 Generalization

The final step of generating our dictionary is to process the opaque mention pairs so that they generalize better. We strip mentions of any determiners, relative clauses, and -ing and -ed clauses. However, we retain adjectives and prepositional modifiers because they are sometimes necessary for coreference to hold (e.g., ***online*** *piracy* and *distribution* ***of pirated material***). We also generalize NEs to their types so that our dictionary entries can function as templates (e.g., *Cook's departure* becomes *<person>'s departure*), but we keep NE tokens that are in the head position as these are pairs containing world knowledge (e.g., *iPad* and *slate*). Finally, we replace all tokens with their lemmas. Table 2 shows a snapshot of the dictionary.

2.5 Semantics of coreference

From manually classifying a sample of 200 dictionary pairs (e.g., Table 2), we find that our dictionary includes many synonymy (e.g., *IPO* and *offering*) and hypernymy relations (e.g., *phone* and *device*), which are the relations that are typically extracted from ontologies for coreference resolution. However, not all synonyms and hypernyms are valid for coreference (recall the *voters-people* vs. *scientists-people* example in the introduction), so our dic-

[8] Report verbs include *argue, claim, say, suggest, tell*, etc.

[9] As a general rule, we extract possessive phrases as subjects (e.g. ***Samsung****'s plan*) and *of*-phrases as objects (e.g. *development of* ***the new logo***).

[10] We can easily detect passive subjects (i-b) as they have their own dependency label, and ergative subjects (ii-b) using a list of ergative verbs extracted from Levin (1993).

(i) a. Developers hacked **the device**.
b. **The device** was hacked.

(ii) a. Police scattered **the crowds**.
b. **The crowds** scattered.

Mention 1	Mention 2
offering	IPO
user	consumer
phone	device
Apple	company
hardware key	digital lock
iPad	slate
content	photo
bug	issue
password	login information
Google	search giant
site	company
filing	complaint
company	government
TouchPad	tablet
medical record	medical file
version	handset
information	credit card
government	chairman
app	software
Android	platform
the leadership change	<person>'s departure
change	update

Table 2: Coreference relations in our dictionary.

tionary only includes the ones that are relevant for coreference (e.g., *update* and *change*). Furthermore, only 3% of our 128,492 opaque pairs are related in WordNet, confirming that our method is introducing a large number of new semantic relations.

We also discover other semantic relations that are relevant for coreference, such as various metonymy relations like mentioning the part for the whole. Again though, we can use some part-whole relations coreferentially (e.g., *car* and *engine*) but not others (e.g., *car* and *window*). Our dictionary includes part-whole relations that have been observed as coreferent at least once (e.g., *company* and *site*). We also extract world-knowledge descriptions for NEs (e.g., *Google* and *the Internet giant*).

3 Integration into a Coreference System

We next integrated our dictionary into an existing coreference resolution system to see if it improves resolution.

3.1 Stanford coreference resolution system

Our baseline is the Stanford coreference resolution system (Lee et al., 2011) which was the highest-scoring system in the CoNLL-2011 Shared Task, and was also part of the highest-scoring system in the CoNLL-2012 Shared Task (Fernandes et al., 2012). It is a rule-based system that includes a total of ten rules (or "sieves") for entity coreference, shown in Table 3. The sieves are applied from highest to lowest precision, each rule extending entities (i.e., mention clusters) built by the previous tiers, but never modifying links previously made. The majority of the sieves rely on string overlap.[11]

Sieve number	Sieve name
1	Discourse processing
2	Exact string match
3	Relaxed string match
4	Precise constructs
5–7	Strict head match
8	Proper head noun match
9	Relaxed head match
10	Pronoun match

Table 3: Rules of the baseline system.

The highly modular architecture made it easy for us to integrate additional sieves using our dictionary to increase recall.

3.2 Dictionary sieves

We propose four new sieves, each one using a different granularity level from our dictionary, with each consecutive sieve using higher precision relations than the previous one. The Dict 1 sieve uses only the heads of mentions in each relation (e.g., *devices*). Dict 2 uses the heads and one premodifier, if it exists (e.g., *iOS devices*). Dict 3 uses the heads and up to two premodifiers (e.g., *new iOS devices*). Dict 4 uses the full mentions, including any postmodifiers (e.g., *new iOS devices for businesses*).

We take advantage of frequency counts to get rid of low-precision coreference pairs and only keep (i) pairs that have been seen more than 75 times (Dict 1) or 15 times (Dict 2, Dict 3, Dict 4); and (ii) pairs with a frequency count larger than 8 (Dict 1) or 2 (Dict 2, Dict 3, Dict 4) and a normalized PMI score larger than 0.18. We use the normalized PMI score (Bouma, 2009) as a measure of association between the mentions m_i and m_j of a

[11] Exceptions: sieve 1 links first-person pronouns inside a quotation with the speaker; sieve 4 links mention pairs that appear in an appositive, copular, acronym, etc., construction; sieve 10 implements generic pronominal coreference resolution.

dictionary pair, computed as

$$(\ln \frac{p(m_i,m_j)}{p(m_i)p(m_j)}) \; / \; -\ln p(m_i, m_j)$$

These thresholds were set on the development set.

Since the different coreference rules in the Stanford system are arranged in decreasing order of precision, we start by applying the sieve that uses the highest-precision relations in the dictionary (Dict 4), followed by Dict 3, Dict 2, and Dict 1. We add these new sieves right before the last sieve, as the pronominal sieve can perform better if opaque mentions have been successfully linked. The current sieves only use the dictionary for linking singular mentions, as the experiments on the dev showed that plural mentions brought too much noise.

For any mention pair under analysis, each sieve checks whether it is supported by the dictionary as well as whether basic constraints are satisfied, such as number, animacy and NE-type agreement, and NE–common noun order (not the opposite).

4 Experiments

4.1 Data

Although our dictionary creation technology can apply across domains, our current coreference dictionary is focused on the technical domain, so we created a coreference labeled corpus in this domain for evaluation. We extracted new data from Techmeme (different from that used to extract the dictionary) to create a development and a test set. It is important to note that we do not need comparable data at this stage. A massive comparable corpus is only needed for mining the coreference dictionary (Section 2); once it is built, it can be used for solving coreference within and across documents.

The annotation was performed by two experts, using the Callisto annotation tool. The development and test sets were annotated with coreference relations following the OntoNotes guidelines (Pradhan et al., 2007). We annotated full NPs (with all modifiers), excluding appositive phrases and predicate nominals. Only premodifiers that were proper nouns or possessive phrases were annotated. We extended the OntoNotes guidelines by also annotating singletons. Table 4 shows the dataset statistics.

Dataset	Stories	Docs	Tokens	Entities	Mentions
Dev	4	27	7837	1360	2279
Test	24	24	8547	1341	2452

Table 4: Dataset statistics: development (dev) and test.

4.2 Evaluation measures

We evaluated using six coreference measures, as they sometimes provide different results and there is no agreement on a standard. We used the scorer of the CoNLL-2011 Shared Task (Pradhan et al., 2011).

- MUC (Vilain et al., 1995). Link-based metric that measures how many links the true and system partitions have in common.
- B^3 (Bagga and Baldwin, 1998). Mention-based metric that measures the proportion of mention overlap between gold and predicted entities.
- CEAF-ϕ_3 (Luo, 2005). Mention-based metric that, unlike B^3, enforces a one-to-one alignment between gold and predicted entities.
- CEAF-ϕ_4 (Luo, 2005). The entity-based version of the above metric.
- BLANC (Recasens and Hovy, 2011). Link-based metric that considers both coreference and non-coreference links.
- CoNLL (Denis and Baldridge, 2009). Average of MUC, B^3 and CEAF-ϕ_4. It was the official metric of the CoNLL-2011 Shared Task.

4.3 Results

We always start from the baseline, which corresponds to the Stanford system with the sieves listed in Table 3. This is the set of sieves that won the CoNLL-2011 Shared Task (Pradhan et al., 2011), and they exclude WordNet.

Table 5 shows the incremental scores, on the development set, for the four sieves that use the dictionary, corresponding to the different granularity levels, from the highest precision one (Dict 4) to the lowest one (Dict 1). The largest improvement is achieved by Dict 4 and Dict 3, as they improve recall (R) without hurting precision (P). R is equivalent to P for CEAF-ϕ_4, and vice versa. The other two sieves increase R further, especially Dict 1, but also decrease P, although the trade-off for the F-score (F1) is still positive. It is the best score, with the exception of B^3.

System	MUC R	MUC P	MUC F1	B^3 R	B^3 P	B^3 F1	CEAF-ϕ_3 R / P / F1	CEAF-ϕ_4 R	CEAF-ϕ_4 P	CEAF-ϕ_4 F1	BLANC R	BLANC P	BLANC B	CoNLL F1
Baseline	55.9	**72.8**	63.3	74.1	**89.8**	81.2	74.6	**85.2**	73.6	79.0	66.6	87.1	72.6	74.5
+Dict 4	57.0	**72.8**	63.9	75.1	89.4	81.6	75.3	**85.2**	74.3	79.4	68.2	**87.3**	74.2	75.0
+Dict 3	57.6	**72.8**	64.3	75.4	89.3	**81.7**	**75.5**	85.1	74.6	79.5	68.4	87.2	**74.4**	**75.2**
+Dict 2	57.6	72.5	64.2	75.4	89.1	**81.7**	75.4	85.0	74.6	79.5	68.4	87.0	74.3	75.1
+Dict 1	**58.4**	71.9	**64.5**	**75.7**	88.5	81.6	**75.5**	84.6	**75.1**	**79.6**	**68.6**	86.6	**74.4**	**75.2**

Table 5: Incremental results for the four sieves using our dictionary on the development set. Baseline is the Stanford system without the WordNet sieves. Scores are on gold mentions.

System	MUC R	MUC P	MUC F1	B^3 R	B^3 P	B^3 F1	CEAF-ϕ_3 R / P / F1	CEAF-ϕ_4 R	CEAF-ϕ_4 P	CEAF-ϕ_4 F1	BLANC R	BLANC P	BLANC B	CoNLL F1
Baseline	62.4	**78.2**	69.4	73.7	**89.5**	80.8	75.1	**86.2**	73.8	79.5	71.4	88.6	77.3	76.6
w/ WN	63.5	75.3	68.9	74.2	87.5	80.3	74.1	83.7	74.1	78.6	71.8	87.3	77.3	75.9
w/ Dict	64.7*	77.6*	70.6*	**75.7***	88.5*	81.6*	**76.5***	85.3*	75.0*	79.9*	**74.6***	88.6	**79.9***	77.3*
w/ Dict + Context	**64.8***	77.8*	**70.7***	**75.7***	88.6*	**81.7***	**76.5***	85.5*	**75.1***	**80.0***	**74.6***	**88.7**	**79.9***	**77.5***

Table 6: Performance on the test set. Scores are on gold mentions. Stars indicate a statistically significant difference with respect to the baseline.

Table 6 reports the scores on the test set and compares the scores obtained by adding the WordNet sieves to the baseline (w/ WN) with those obtained by adding the dictionary sieves (w/ Dict). Whereas adding WordNet only brings a small improvement in R that is much lower than the loss in P, the dictionary sieves succeed in increasing R by a larger amount and at a smaller cost to P, resulting in a significant improvement in F1: 1.2 points according to MUC, 0.8 points according to B^3, 1.4 points according to CEAF-ϕ_3, 0.4 points according to CEAF-ϕ_4, 2.6 points according to BLANC, and 0.7 points according to CoNLL. Section 5.2 presents the last line (w/ Dict + Context).

5 Discussion

5.1 Error analysis

Thanks to the dictionary, the coreference system improves the baseline by establishing coreference links between the bolded mentions in (3) and (4).

(3) With **Groupon Inc.**'s stock down by half from its IPO price and **the company** heading into its first earnings report since an accounting blowup [...] outlining opportunity ahead and the promise of new products for **the daily-deals company**.

(4) Thompson revealed the diagnosis as evidence arose that seemed to contradict his story about why he was not responsible for a degree listed on his resume that he does not have, the newspaper reports, citing anonymous sources familiar with **the situation** [...] a Yahoo board committee appointed to investigate **the matter**.

The first case requires world knowledge and the second case, semantic knowledge.

We manually analyzed 40 false positive errors caused by the dictionary sieves. Only a small number of them were due to noise in the dictionary. The majority of errors were due to the discourse context: the two mentions could be coreferent, but not in the given context. For example, *Apple* and *company* are potentially coreferent—which is successfully captured by our dictionary—and while they are coreferent in (5), they are not in (6).[12]

(5) It will only get better as **Apple** will be updating it with iOS6, an operating system that **the company** will likely be showing off this summer.

(6) Since *Apple* reinvented the segment, **Microsoft** is the latest entrant into the tablet market, banking on its Windows 8 products to bridge the gap between PCs and tablets. [...] **The company** showed off Windows 8 last September.

[12] Examples in this section show gold coreference relations in bold and incorrectly predicted coreferent mentions in italics.

In these cases it does not suffice to check whether the opaque mention pair is included in the coreference dictionary, but we need a method for taking the surrounding context into account. In the next section we present our preliminary work in this direction.

5.2 Context fit

To help the coreference system choose the right antecedent in examples like (6), we exploit the fact that *the company* is closely followed by *Windows 8*, which is a clue for selecting *Microsoft* instead of *Apple* as the antecedent. We devise a contextual constraint that rules out a mention pair if the contexts are incompatible. To check for context compatibility, we borrow the idea of topic signatures from Lin and Hovy (2000) and that Agirre et al. (2001) used for Word Sense Disambiguation. Instead of identifying the keywords of a topic, we find the NEs that tend to co-occur with another NE. For example, the signature for *Apple* should include terms like *iPhone*, *MacBook*, *iOS*, *Steve Jobs*, etc. This is what we call the **NE signature** for *Apple*.

To construct NE signatures, we first compute the log-likelihood ratio (LLR) statistic between NEs in our corpus (the same one used to build the dictionary). Then, the signature for a NE, w, is the list of k other NEs that have the highest LLR with w. The LLR between two NEs, w_1 and w_2, is $-2 \ln \frac{L(H_1)}{L(H_2)}$, where H_1 is the hypothesis that
$P(w_1 \in sent | w_2 \in sent) = P(w_1 \in sent | w_2 \notin sent)$,
H_2 is the hypothesis that
$P(w_1 \in sent | w_2 \in sent) \neq P(w_1 \in sent | w_2 \notin sent)$,
and $L(\cdot)$ is the likelihood. We assume a binomial distribution for the likelihood.

Once we have NE signatures, we determine the context fit as follows. When the system compares a NE antecedent with a (non-NE) anaphor, we check whether any NEs in the anaphor's sentence are in the antecedent's signature. We also check whether the antecedent is in the signature list of any NE's in the anaphor's sentence. If neither of these is true, we do not allow the system to link the antecedent and the anaphor. In (6), *Apple* is not linked with *the company* because it is not in Windows' signature, and *Windows* is not in Apple's signature either (but *Microsoft* is in Windows' signature).

The last two lines in Table 6 compare the scores without using this contextual feature (w/ Dict) with those using context (w/ Dict + Context). Our feature for context compatibility leads to a small but positive improvement, taking the final improvement of the dictionary sieves to be about 1 percentage point above the baseline according to all six evaluation measures. We leave as future work to test this idea on a larger test set and refine it further so as to address more challenging cases where comparing NEs is not enough, like in (7).

(7) **Snapchat** will notify users [...] The program is available for free in *Apple*'s App Store [...] While **the company** "attempts to delete image data as soon as possible after the message is transmitted," it cannot guarantee messages will always be deleted.

To resolve (7), it would be helpful to know that Snapchat is a picture messaging platform, as the context mentions *image data* and *messages*.

6 Related Work

Existing ontologies are not optimal for solving opaque coreferent mentions because of both a precision and a recall problem (Lee et al., 2011; Uryupina et al., 2011). On the other hand, using data-driven methods such as distributional semantics for coreference resolution suffers especially from a precision problem (Ng, 2007). Our work combines ideas from distributional semantics and paraphrase acquisition methods in order to efficiently use contextual information to extract coreference relations.

The main idea that we borrow from paraphrase acquisition is the use of monolingual (non-parallel) comparable corpora, which have been exploited to extract both sentence-level (Barzilay and McKeown, 2001) and sub-sentential-level paraphrases (Shinyama and Sekine, 2003; Wang and Callison-Burch, 2011). To ensure that the NPs are coreferent, we limit the meaning of *comparable corpora* to collections of documents that report on the very same story, as opposed to collections of documents that are about the same (general) topic. However, the distinguishing factor is that while most paraphrasing studies, including Lin and Pantel (2001), use NEs—or nouns in general—as pivots to learn paraphrases of their surrounding context, we use verbs as pivots to learn coreference relations at the NP level.

There are many similarities between paraphrase and coreference, and our work is most similar to

that by Wang and Callison-Burch (2011). However, some paraphrases that might not be considered to be valid (e.g., *under $200* and *around $200*) can be acceptable coreference relations. Unlike Wang and Callison-Burch (2011), we do not work on document pairs but on sets of at least five (comparable) documents, and we do not require sentence alignment, but just verb alignment.

Another source of inspiration is the work by Bean and Riloff (2004). They use contextual roles (i.e., the role that an NP plays in an event) for extracting patterns that can be used in coreference resolution, showing the relevance of verbs in deciding on coreference between their arguments. However, they use a very small corpus (two domains) and do not aim to build a dictionary. The idea of creating a repository of extracted concept-instance relations appears in Fleischman et al. (2003), but restricted to person-role pairs, e.g. *Yasser Arafat* and *leader*. Although it was originally designed for answering who-is questions, Daumé III and Marcu (2005) successfully used it for coreference resolution.

The coreference relations that we extract might overlap but go beyond those detected by Bansal and Klein (2012)'s Web-based features. First, they focus on NP headwords, while we extract full NPs, including multi-word mentions. Second, the fact that they use the Google *n*-gram corpus means that the two headwords must appear at most four words apart, thus ruling out coreferent mentions that can only appear far from each other. Finally, while their extraction patterns focus on synonymy and hypernymy relations, we discover other types of semantic relations that are relevant for coreference (Section 2.5).

7 Conclusions

We have pointed out an important problem with current coreference resolution systems: their heavy reliance on string overlap. Pronouns aside, opaque mentions account for 65% of the errors made by state-of-the-art systems. To improve coreference scores beyond 60-70%, we therefore need to make better use of semantic and world knowledge to deal with non-identical-string coreference. But, as we have also shown, coreference is not the same as semantic similarity or hypernymy. Only certain semantic relations in certain contexts are good cues for coreference. We therefore need semantic resources specifically targeted at coreference.

We proposed a new solution for detecting opaque mention pairs: restricting distributional similarity to a comparable corpus of articles about the very same story, thus ensuring that similar mentions will also likely be coreferent. We used this corpus to build a dictionary focused on coreference, and successfully extracted the specific semantic and world knowledge relevant for coreference. The resulting dictionary can be added on top of any coreference system to increase recall at a minimum cost to precision. Integrated into the Stanford coreference resolution system, which won the CoNLL-2011 shared task, the F-score increases about 1 percentage point according to all of the six evaluation measures. The dictionary and NE signatures are available on the Web.[13]

We showed that apart from the need for extracting coreference-specific semantic and world knowledge, we need to take into account the context surrounding the mentions. The results from our preliminary work for identifying incompatible contexts is promising.

Our unsupervised method for extracting opaque coreference relations can be easily extended to other domains by using online news aggregators, and trained on more data to build a more comprehensive dictionary that can increase recall even further. We integrated the dictionary into a rule-based coreference system, but it remains for future work to integrate it into a learning-based architecture, where the system can combine the dictionary features with other features. This can also make it easier to include contextual features that take into account how well a dictionary pair fits in a specific context.

Acknowledgments

We would like to thank the members of the Stanford NLP Group, Valentin Spitkovsky, and Ed Hovy for valuable comments at various stages of the project.

The first author was supported by a Beatriu de Pinós postdoctoral scholarship (2010 BP-A 00149) from Generalitat de Catalunya. We also gratefully acknowledge the support of Defense Advanced Research Projects Agency (DARPA) Machine Reading Program under Air Force Research Laboratory (AFRL) prime contract no. FA8750-09-C-0181.

[13] http://nlp.stanford.edu/pubs/coref-dictionary.zip

References

Eneko Agirre, Olatz Ansa, David Martinez, and Eduard Hovy. 2001. Enriching wordnet concepts with topic signatures. In *Proceedings of the NAACL Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, pages 23–28.

Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *Proceedings of the LREC 1998 Workshop on Linguistic Coreference*, pages 563–566.

Mohit Bansal and Dan Klein. 2012. Coreference semantics from web features. In *Proceedings of ACL*, pages 389–398.

Regina Barzilay and Kathleen McKeown. 2001. Extracting paraphrases from a parallel corpus. In *Proceedings of ACL*, pages 50–57.

David Bean and Ellen Riloff. 2004. Unsupervised learning of contextual role knowledge for coreference resolution. In *Proceedings of NAACL-HTL*.

Geolof Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction. In *Proceedings of the Biennial GSCL Conference*, pages 31–40.

Hal Daumé III and Daniel Marcu. 2005. A large-scale exploration of effective global features for a joint entity detection and tracking model. In *Proceedings of HLT-EMNLP*, pages 97–104.

Pascal Denis and Jason Baldridge. 2009. Global joint models for coreference resolution and named entity classification. *Procesamiento del Lenguaje Natural*, 42:87–96.

Eraldo Fernandes, Cícero dos Santos, and Ruy Milidiú. 2012. Latent structure perceptron with feature induction for unrestricted coreference resolution. In *Proceedings of CoNLL - Shared Task*, pages 41–48.

Michael Fleischman, Eduard Hovy, and Abdessamad Echihabi. 2003. Offline strategies for online question answering: answering questions before they are asked. In *Proceedings of ACL*, pages 1–7.

Aria Haghighi and Dan Klein. 2010. Coreference resolution in a modular, entity-centered model. In *Proceedings of HLT-NAACL*, pages 385–393.

Hamidreza Kobdani, Hinrich Schütze, Michael Schiehlen, and Hans Kamp. 2011. Bootstrapping coreference resolution using word associations. In *Proceedings of ACL*, pages 783–792.

Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2011. Stanford's multi-pass sieve coreference resolution system at the CoNLL-2011 Shared Task. In *Proceedings of CoNLL - Shared Task*, pages 28–34.

Beth Levin. 1993. *English Verb Class and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago.

Chin-Yew Lin and Eduard Hovy. 2000. The automated acquisition of topic signatures for text summarization. In *Proceedings of COLING*, pages 495–501.

Dekang Lin and Patrick Pantel. 2001. DIRT - Discovery of inference rules from text. In *Proceedings of the ACM SIGKDD*, pages 323–328.

Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In *Proceedings of HLT-EMNLP*, pages 25–32.

Vincent Ng. 2007. Shallow semantics for coreference resolution. In *Proceedings of IJCAI*, pages 1689–1694.

Joakim Nivre, Johan Hall, and Jens Nilsson. 2004. Memory-based dependency parsing. In *Proceedings of CoNLL*, pages 49–56.

Simone Paolo Ponzetto and Michael Strube. 2006. Exploiting semantic role labeling, WordNet and Wikipedia for coreference resolution. In *Proceedings of HLT-NAACL*, pages 192–199.

Sameer S. Pradhan, Lance Ramshaw, Ralph Weischedel, Jessica MacBride, and Linnea Micciulla. 2007. Unrestricted coreference: Identifying entities and events in OntoNotes. In *Proceedings of ICSC*, pages 446–453.

Sameer Pradhan, Lance Ramshaw, Mitchell Marcus, Martha Palmer, Ralph Weischedel, and Nianwen Xue. 2011. CoNLL-2011 Shared Task: Modeling unrestricted coreference in OntoNotes. In *Proceedings of CoNLL - Shared Task*, pages 1–27.

Altaf Rahman and Vincent Ng. 2011. Coreference resolution with world knowledge. In *Proceedings of ACL*, pages 814–824.

Marta Recasens and Eduard Hovy. 2010. Coreference resolution across corpora: Languages, coding schemes, and preprocessing information. In *Proceedings of ACL*, pages 1423–1432.

Marta Recasens and Eduard Hovy. 2011. BLANC: Implementing the Rand index for coreference evaluation. *Natural Language Engineering*, 17(4):485–510.

Yusuke Shinyama and Satoshi Sekine. 2003. Paraphrase acquisition for information extraction. In *Proceedings of ACL*, pages 65–71.

Veselin Stoyanov, Nathan Gilbert, Claire Cardie, and Ellen Riloff. 2009. Conundrums in noun phrase coreference resolution: Making sense of the state-of-the-art. In *Proceedings of ACL-IJCNLP*, pages 656–664.

Olga Uryupina, Massimo Poesio, Claudio Giuliano, and Kateryna Tymoshenko. 2011. Disambiguation and filtering methods in using web knowledge for coreference resolution. In *Proceedings of FLAIRS*, pages 317–322.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. In *Proceedings of MUC-6*, pages 45–52.

Rui Wang and Chris Callison-Burch. 2011. Paraphrase fragment extraction from monolingual comparable corpora. In *Proceedings of the 4th ACL Workshop on Building and Using Comparable Corpora*, pages 52–60.

Xiaofeng Yang and Jian Su. 2007. Coreference resolution using semantic relatedness information from automatically discovered patterns. In *Proceedings of ACL*, pages 528–535.