

Disambiguation and Filtering Methods in Using Web Knowledge for Coreference Resolution

Olga Uryupina and Massimo Poesio

CiMeC, University of Trento
 uryupina@gmail.com, massimo.poesio@unitn.it

Claudio Giuliano and Kateryna Tymoshenko

Fondazione Bruno Kessler
 giuliano@fbk.eu, tymoshenko@fbk.eu

Abstract

We investigate two publicly available web knowledge bases, Wikipedia and Yago, in an attempt to leverage semantic information and increase the performance level of a state-of-the-art coreference resolution (CR) engine. We extract semantic compatibility and aliasing information from Wikipedia and Yago, and incorporate it into a CR system. We show that using such knowledge with no disambiguation and filtering does not bring any improvement over the baseline, mirroring the previous findings (Ponzetto and Poesio 2009). We propose, therefore, a number of solutions to reduce the amount of noise coming from web resources: using disambiguation tools for Wikipedia, pruning Yago to eliminate the most generic categories and imposing additional constraints on affected mentions. Our evaluation experiments on the ACE-02 corpus show that the knowledge, extracted from Wikipedia and Yago, improves our system's performance by 2-3 percentage points.

Introduction

Coreference resolution is an essential prerequisite for a variety of NLP tasks: information extraction, machine translation, summarization and many others. State-of-the-art resolvers often rely on very shallow features. Coreference, however, is a complex phenomenon and therefore a robust and reliable approach to the problem should address numerous linguistic and common-sense aspects. Previous studies have investigated possibilities for extracting such knowledge from WordNet (Harabagiu, Bunescu, and Maiorano 2001; Huang et al. 2009), Wikipedia (Ponzetto and Strube 2006) or large text corpora (Haghighi and Klein 2009; Bean and Riloff 2004; Garera and Yarowsky 2006; Yang and Su 2007). Still, knowledge acquisition remains a bottleneck for state-of-the-art CR algorithms.

The Semantic Web made available a large amount of information, which constitute a valuable source of semantics. However, it's difficult to integrate them with state-of-the-art coreference methods: different SW resources use different schemes; knowledge bases have irregular coverage; SW knowledge is encoded in logical form while coreference systems are based on statistical models.

Copyright © 2011, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

We investigate two web knowledge bases, Wikipedia and Yago, in our attempt to leverage aliasing and semantic information and thus increase the performance level of our CR engine. Wikipedia is a collaborative encyclopedic resource with more than 3 million entries for English. Yago is a knowledge base, linking Wikipedia entries to the WordNet ontology. Yago ontology contains 1 million entities and 5 million facts. It supports various semantic relations among concepts, but in this study we only focus on the *means* and *type* relations: the former encodes synonymy and the latter – hyperonymy and categorial information.

Previous studies show that, even though at earlier stages web knowledge bases might be a source of valuable information (Ponzetto and Strube 2006), the expansion of such resources inevitably leads to an increase in the amount of noise, making them hardly usable for our application (Ponzetto and Poesio 2009).

In our study we extract semantic compatibility and aliasing information from Wikipedia and Yago, and incorporate it into a state-of-the-art CR system. We show that a naive approach does not bring any improvement over the baseline, mirroring the previous findings. We propose, however, a number of solutions to reduce the amount of noise coming from web resources: using disambiguation tools for Wikipedia, pruning Yago to eliminate the most populated categories and imposing additional constraints on affected mentions. Our evaluation experiments on the ACE-02 corpus show that the knowledge, extracted from Wikipedia and Yago, improves our system's performance by 2-3 percentage points. We also perform an error analysis, identifying cases where semantic compatibility information induced from the web leads to spurious coreference links.

Baseline

For our experiments, we use BART (Versley et al. 2008b), a modular toolkit for coreference resolution that supports state-of-the-art statistical approaches to the task and enables efficient feature engineering.

We view coreference resolution as a binary classification problem. Each classification instance consists of two mentions, i.e. an anaphor and its potential antecedent. Instances are modeled as feature vectors (cf. Table 1, upper part) and are handed over to a binary classifier that decides, whether the anaphor and the candidate antecedent are coreferent or

not. All the feature values are computed automatically.

Basic feature set
MentionType(M_i), MentionType(M_j)
IsCoordination(M_i), IsCoordination(M_j)
SemanticClass(M_i), SemanticClass(M_j)
GenderAgreement(M_i, M_j)
NumberAgreement(M_i, M_j)
AnimacyAgreement(M_i, M_j)
StringMatch(M_i, M_j)
Alias(M_i, M_j)
Apposition(M_i, M_j)
FirstMention(M_i)
Distance(M_i, M_j)
Wikipedia and Yago features
Wiki-Alias(M_i, M_j)
Yago-Means(M_i, M_j)
Yago-Type(M_i, M_j)

Table 1: Features used by BART: each feature describes a pair of mentions $\{M_i, M_j\}$, $i < j$, where M_i is a candidate antecedent and M_j is a candidate anaphor

We train a maximum entropy classifier and follow the model of (Soon, Ng, and Lim 2001) to partition mentions into coreference sets given the classifier’s decisions.

Our evaluation experiments have been performed on the ACE-02 corpus. A small subset of the training part has been reserved for development. To allow for a better comparison with the state-of-the-art, we show results both for gold (Table 2) and automatically extracted (Table 3) mentions.

The first rows of Tables 2 and 3 show the baseline performance. Note that our baseline is already a very competitive system: it yields the results comparable or superior to other state-of-the-art approaches to the task¹.

Using Wikipedia to improve aliasing

Coreference resolution for named entities can be seen as a separate problem. Several algorithms have been proposed to address this subtask specifically (McCallum and Wellner 2003). Some coreference links between named entities, mainly LOCATIONS, can be accounted for by simple string matching. Names of PERSONS and ORGANIZATIONS, however, often require a more complex *aliasing* strategy:

- (1) The ‘94th World Series opens tomorrow in New York with the favored Yankees taking on [the upstart San Diego Padres]. <..>
Ken Belsan says that while many Japanese will be pulling for the underdog-based Stars, they’re not inclined to pull for [the upstart Padres].

Even though it is clear to a human reader that the two mentions refer to the same entity, most coreference resolvers

¹A comparison of different systems by (Poon and Domingos 2008) suggests the state-of-the-art performance on gold mentions for ACE02-BNEWS and ACE02-NWIRE at 68-69% and 67% MUC-F1 respectively.

cannot reliably establish this link. A conservative aliasing algorithm would check whether the anaphor is a substring of the antecedent, and thus would not be able to capture this link. Less conservative approach, relying, for example, on a match between the last tokens or heads of two mentions, would handle this pair correctly, but, as a drawback, they would also cluster names of relatives, such as “Bill Clinton” and “Hillary Clinton”, into the same entity. Accurate aliasing techniques for coreference resolution rely on sophisticated inference for guessing and comparing names’ structures (cf., for example, (Versley et al. 2008a; Uryupina 2004)).

In the present study we use Wikipedia to improve our aliasing algorithm. We link each mention to its Wikipedia page, thus providing a normalized version of each named entity in a document. We then compare the page ids to determine mentions with the same alias. For the example above, both mentions are linked in Wikipedia to San Diego Padres and thus are considered coreferent by our `wiki-aliasing` feature.

We extend the scope of our `wiki-alias` feature to all the mentions, in an attempt to account for synonymy (for example, “kids” and “children” link to the same Wikipedia entry `Child`). Below we also investigate another source of the synonymy information, the Yago *means* relation.

Incorporating Wikipedia knowledge as a feature

In order to acquire common-sense information from web knowledge bases, we have to link mentions to corresponding Wikipedia entries. This can be achieved in a straightforward way: for each mention, we extract its minimal span² and then query the Wikipedia. This approach gives us, on the one hand, a starting point for extracting more knowledge describing a particular mention (i.e. its Wiki entry) and, on the other hand, a wiki-based aliasing feature (the `wiki-alias` feature is set to 1 when the anaphor and the antecedent share the same Wiki entry). Our analysis of the development set has revealed two major issues with the naive approach:

- Pronouns are not covered by Wikipedia and therefore might receive spurious entries: for example, “who” is linked to `The Who` and “he” is linked to `Helium`.
- Some mentions have multiple entries and our naive approach would just pick the first one from the list. This again decreases the system’s accuracy.

The former issue can be tackled by imposing an additional constraint on the mention types for `wiki-aliasing`. The latter problem, however, requires more sophisticated machinery, described below.

Disambiguating mentions into Wikipedia senses

The problem is casted as a WSD exercise, in which each mention in a document (excluding pronouns) has to be disambiguated using Wikipedia to provide sense inventory and

²Roughly speaking, the minimal span corresponds to the head for nominal mentions and the whole name for NEs, cf. ACE guidelines for more details.

the training data. To create the training set, for each mention M we collect from the Wikipedia dump all the contexts where M is an anchor of an internal link. The set of target articles represents the senses of M and the contexts are used as labeled training examples.

To disambiguate mentions in text, we implemented a kernel-based approach similar to the one proposed by (Giuliano, Gliozzo, and Strapparava 2009). Different kernel functions are employed to integrate syntactic, semantic, and pragmatic knowledge sources typically used in the WSD literature. The strategy adopted by kernel methods consists of splitting the learning problem into two parts. They first embed the input data in a suitable feature space, and then use a linear algorithm (e.g., support vector machines) to discover nonlinear patterns in the input space. The kernel function is the only task-specific component of the learning algorithm. For each knowledge source, a specific kernel has been defined. By exploiting the property of kernels, basic kernels are then combined to define the WSD kernel. Specifically, we use a combination of gap-weighted subsequences, bag-of-words, and latent semantic kernels (Shawe-Taylor and Cristianini 2004).

Gap-weighted subsequences kernel. This kernel learns syntactic and associative relations between words in a local context. We extended the gap-weighted subsequences kernel to subsequences of word forms, stems, part-of-speech tags, and orthographic features (capitalization, punctuation, numerals, etc.). We defined gap-weighted subsequences kernels to work on subsequences of length up to 5.

Bag-of-words kernel. This kernel learns domain, semantic, topical information. Bag-of-words kernel takes as input a wide context window around the target mention. Words are represented using stems.

Latent semantic kernel. This kernel extracts semantic information through co-occurrence analysis in a corpus. The technique used to extract the co-occurrence statistics relies on a singular value decomposition of the term-by-document matrix. The latent semantic model is derived from the 200,000 most visited Wikipedia articles, after removing terms that occur less than 5 times, the resulting dictionary contain about 300,000 and 150,000 terms respectively. We used the SVDLIBC package to compute the SVD, truncated to 400 dimensions.³ To classify each mention in Wikipedia entries, we used a LIBSVM package.⁴ No parameter optimization was performed.

This machinery allows us to define an improved version of our Wikipedia-based aliasing: the `wiki-alias*` feature is set to 1 if and only if both (a) the anaphor and the antecedent are non-pronominal mentions and (b) their disambiguated Wikipedia entries are the same.

Using Yago to extract semantic knowledge

The most important entities are mentioned many times throughout a document, realized differently on the surface level: as names, pronouns or nominals. For less-known entities, such re-descriptions can be introduced explicitly:

³<http://tedlab.mit.edu/~dr/svdlbc/>

⁴<http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

- (2) "The emperor has no coattails," said [Matthew Miller], [the spokesman for the state Democratic Party's campaign arm, the Campaign for Connecticut Families].

Such coreference links can be accounted for by designing syntactic features, covering appositive and copula constructions. Consider, however, the following examples:

- (3) Senior U.S. officials are quoted in the "[Far Eastern Economic Review]." [The publication] says military chiefs in the U.S. and South Korea still have to approve the revised strategy
- (4) However, I have spoken to many, many people who are emigres from [Afghanistan] and are trying desperately to do something about the plight of [their country], and a lot of people that I've spoken to work with the women in the refugee camps who fled Afghanistan after the takeover and are now living in camps in Pakistan.

In this examples, the documents provide no syntactic clue for the resolution of the two anaphors, "the publication" and "their country". The reader should rely on common-sense knowledge to deduce that "Far Eastern Economic Review" is an instance of "publication" and "Afghanistan" is an instance of "country". Obviously, a coreference resolver based on shallow features cannot account for such links.

Several attempts have been made in the literature to extract this information from publicly available sources, mainly from the WordNet. These approaches suffer from two major problems. First, the WordNet coverage, especially for proper names, is not sufficient. For example, it provides an extensive list of countries, but not cities. The proper name "Far Eastern Economic Review" is not covered either. Second, the senses are too fine-grained and thus the hierarchy is often difficult to use in any reliable fashion. For example, only the sixth sense of "review" is connected to "publication". Therefore we either need a complex disambiguation machinery or some noise-reduction mechanisms to be able to use WordNet for coreference resolution.

In this study we extract the relevant knowledge from Yago (Suchanek, Kasneci, and Weikum 2007) – an ontology extracted from Wikipedia and unified with WordNet. The information we extract in this experiment concerns only the *means* and *type* facts about Yago concepts. Previous studies on using Yago for coreference include (Bryl et al. 2010a) and (Bryl et al. 2010b). However, they do not investigate filtering techniques and provide evaluation results only on a subset of mentions.

Incorporating Yago knowledge as features

As before, we start with a naive approach to the task. For a named entity, we use the (disambiguated, cf. above) Wikipedia entry to link it to a Yago node. For nominals, we extract the head nouns and use them as Yago entries⁵.

We define two features, `yago-means` and `yago-type` as follows. The `yago-means` feature is set to true if and only if the Yago entries for the anaphor and the antecedent

⁵At the moment, Yago does not support non-NE concepts represented in Wikipedia.

stay in the *means* relation to the same entity (we add reflexive *means* relations from each node to itself). The *yago-type* feature is set to true if and only if (a) there exist Yago nodes T_1 and T_2 such as the anaphor stays in *type* relation with T_1 and (b) the antecedent stays in *means* relation with T_2 and (c) T_1 and T_2 stay in *type* relation or vice versa. Roughly speaking, the former feature should encode the aliasing and synonymy information, and the latter one – hyperonymy⁶. Figure 1 illustrates our patterns for extracting the Yago features.

We have tested the accuracy of our Yago-based features on the development data to identify a number of problematic cases with the *yago-type* feature:

- Discourse new descriptions (for example, “another city”) are classified as hyperonyms of some non-corefering preceding mentions
- Mentions with too generic head nouns (for example, “a group”) are classified as hyperonyms of virtually any candidate antecedent
- When a named entity is linked to its preceding hyperonym, a spurious chain might arise (for example, [“New York”, “city”, “Los Angeles”]). This is a drawback of our very local resolution algorithm.

Still, the *yago-type* feature encodes valuable knowledge that is otherwise not available to the system – recall the examples of “publication” and “their country”. We have therefore implemented a number of filtering solutions to be able to leverage this information without too much noise.

Filtering

We impose a number of constraints on the information extracted from Yago to get less noisy hyperonymy feature (*yago-type**).

Discourse new description as hyperonyms. A mention’s head noun may be a hyperonym of some candidate antecedent, but the context may explicitly indicate that the two are not coreferent:

- (5) [India]’s advantage, it simply has more skilled, English-speaking programmers than [any other country] outside the U.S.

Here the choice of a determiner indicates that the two mentions stay in a discourse relation, but it is not coreference.

To account for such pairs, we have extracted from the ACE corpus a set of determiners characteristic for mentions with no antecedents (“another”, “no” etc). We do not rely on a more complex discourse new detector (cf. (Poesio et al. 2004) for an overview of relevant approaches) for efficiency reasons. If the anaphor has such a determiner, we reset our Yago features to false, regardless of their original values.⁷

⁶We use the term “hyperonymy” in a broad sense here: for a common noun, a hyperonym is any predecessor node in an Is-A hierarchy (for example, “city” is a hyperonym of “capital”), for a named entity, a hyperonym is any appropriate category (both “capital” and “city” are hyperonyms of “Washington”).

⁷The system cannot learn such information due to the specifics of the (Soon, Ng, and Lim 2001) algorithm: discourse new mentions do not produce any training examples.

Too common hyperonyms. Some mentions have head nouns that can be considered a hyperonym of virtually any other mention. We have collected statistics, measuring for each Yago node the total number of its instances and hyperonyms among the ACE mentions. We have then manually analyzed the highest ranking nodes to identify three groups:

- Some head nouns, for example, “country” are indeed often used as hyperonyms or category terms. This reflects the fact that some common-sense knowledge is expected to be shared by virtually all the readers and is therefore often used to produce re-descriptions.
- Some head nouns, such as “group” or “part” are almost never used as hyperonyms or category terms of some anaphoric preceding mentions in the ACE documents. These nouns are too generic and using them as anaphoric descriptions would lead to too much ambiguity. Typically, when such a mention is an anaphor, the link is marked explicitly, as in “These latter-born infants were 10 times as likely to be killed as were infants in [the lowest risk group]: [firstborn babies of mothers 25 and older.]”.
- Finally, some head nouns, such as “area”, are typically use with pre- or post-modifiers crucial for their correct interpretation. Thus, “Los Angeles area” and “Woodland Hills” are not compatible, even though “Woodland Hills” is an instance of “area”.

This analysis helps us create a stop-list of Yago nodes that should not trigger the *yago-type* feature, including cases similar to “group” or “part”, but neither “country” nor “area”.

Globally inconsistent decisions. Our baseline system relies on a very local resolution strategy. This leads to spurious coreference chains when the *yago-type* feature is activated: by resolving, for example, first “the city” to “New York” and then “San Francisco” to “the city” we merge all the three mentions into one chain. We have investigated a mention-ranking algorithm to alleviate the problem, but the results were not reliable enough. Therefore we prohibit any *yago-type* links for pairs of mention where the anaphor is a named entity and the antecedent is a common noun phrase. This constraint is motivated by our system architecture and we hope to eliminate it in the future by adopting a more global framework.

All these filtering solutions help us create new, cleaner features, *yago-means** and *yago-type**. In the following section we will see whether they lead to any improvement in the performance level.

Evaluation and Error Analysis

Our evaluation results on the ACE-02 test set (Tables 2 and 3) show that the two web knowledge bases help improve the performance level of a coreference resolution system by 2-3 percentage points, provided extra measures are taken to control for spurious links.

The Wikipedia knowledge (*wiki-alias*) yields around 1% improvement. Note that our system already makes use of an advanced manually crafted feature for aliasing (cf. Table 1), so this improvement accounts for truly non-trivial cases

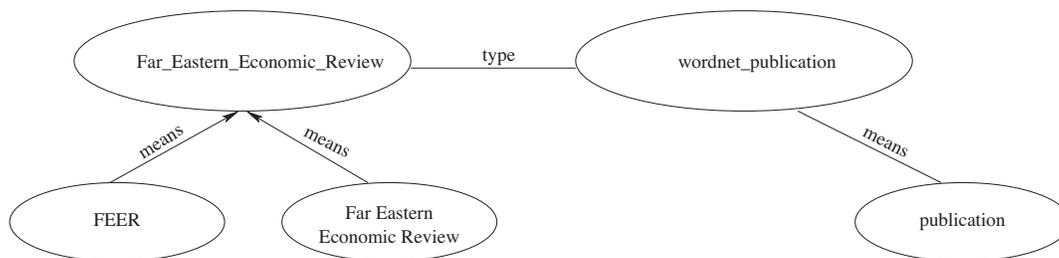


Figure 1: Encoding Yago information: the `yago-means` feature is set to true for [“FEER”, “Far Eastern Economic Review”], the `yago-type` feature is set to true for [“FEER”, “The publication”] and [“Far Eastern Economic Review”, “The publication”]

	NPAPER		NWIRE		BNEWS	
	MUC-F1	CEAF-F1	MUC-F1	CEAF-F1	MUC-F1	CEAF-F1
Baseline	70.3	68.3	70.3	71.8	72.3	70.7
Baseline+wiki	71.6	67.2	71.0	70.2	74.3	72.2
Baseline+wiki*	71.7	67.6	72.0	72.8	75.5	72.8
Baseline+wiki*+yago	71.9	66.8	71.9	72.3	74.9	71.9
Baseline+wiki*+yago*	72.2	67.9	72.5	72.3	75.4	72.5

Table 2: System performance with and without Wikipedia and Yago-based features, MUC and CEAF- ϕ_4 F-scores on the ACE-02 corpus with *gold* mentions. Runs with disambiguation and prefiltering marked with *. Significant improvements of the final system, Baseline+wiki*+yago*, over the baseline are shown in boldface (sign test, $p < 0.05$).

of aliasing. The disambiguation machinery yields better results for NPAPER and BNEWS, but not NWIRE.

Yago features do not lead to any improvement when the hyperonymy information is used without any prefiltering: even though our system obtains valuable knowledge, it is too noisy to affect the classification in any positive way. With a number of task-specific adjustments, however, the Yago knowledge yields a consistent improvement over the baseline for all the three domains.

We have performed a manual error analysis for the BNEWS domain, comparing our baseline to the most advanced (Baseline+wiki*+yago*) setting. Of the 72 spurious links introduced by our Wikipedia and Yago-based features, no single error can be traced back to inconsistencies in the databases. Nine links could have been avoided by elaborating on already suggested filtering techniques (e.g., by extending the stop-list).

Around one third of all the erroneous links (27 in total) could have been ruled out by accounting for modification. Only six of these cases, however, require complex reasoning (for example, “American women” and “feminist-oriented young women” cannot refer to the same set of women, even though the modifiers are not mutually incompatible). The majority of cases are more straightforward: for example, “the gentleman from California Mr. Rogan” is erroneously resolved to “Gentleman from Wisconsin”.

Finally, for 36 errors, the surface forms of the mentions contain no indication that they are not coreferent – the relevant information can only be extracted from the context:

- (6) The Metchiar government left Slovokia increasingly isolated from its neighbors in Central Europe – Poland, Hungary, and [the Czech Republic]. The incoming

prime minister faces formidable challenges in turning [his country] into an effective market democracy.

The system has linked “his country” to the closest instance of “country”, “the Czech republic”, relying on the `yago-type` feature. The correct antecedent, however, is “Slovakia”, which does not even trigger the `yago-type` feature because of the spelling error. A possible solution in this case would be to combine common-sense knowledge, extracted from the web, with global salience measures.

Conclusion

In this paper we have investigated two publicly available web knowledge bases, Wikipedia and Yago, in an attempt to leverage semantic information and increase the performance level of a state-of-the-art coreference resolution engine.

We show that web knowledge bases might help to improve the performance level of a CR system by 2-3 percentage points, provided disambiguation and prefiltering techniques are implemented to control for spurious links.

This brings us to the conclusion that web knowledge bases are a source of valuable information, that is useful for coreference resolution. However, an extra care should be taken when using such information: a naive approach does not bring any improvement, whereas a few very simple disambiguation and filtering solutions lead to better results.

Our error analysis suggests that the errors introduced by such an approach are not caused by any deficiencies in the web knowledge bases, but reflect the complex nature of the coreference resolution task. We plan therefore to investigate the interaction of the common-sense knowledge (as extracted from Wikipedia and Yago) with syntactic and salience features.

	NPAPER		NWIRE		BNEWS	
	MUC-F1	CEAF-F1	MUC-F1	CEAF-F1	MUC-F1	CEAF-F1
Baseline	65.3	51.7	60.9	55.2	60.7	56.2
Baseline+wiki	65.5	52.4	61.0	55.1	60.9	56.4
Baseline+wiki*	64.9	51.2	60.8	55.7	61.1	56.7
Baseline+wiki*+yago	65.9	51.6	61.6	56.0	62.4	57.2
Baseline+wiki*+yago*	65.1	50.9	61.1	56.8	62.7	57.8

Table 3: System performance with and without Wikipedia and Yago-based features, MUC and CEAF- ϕ_4 F-scores on the ACE-02 corpus with *system* mentions. Runs with disambiguation and prefiltering marked with *. Significant improvements of the final system, Baseline+wiki*+yago*, over the baseline are shown in boldface (sign test, $p < 0.05$).

References

- Bean, D. L., and Riloff, E. 2004. Unsupervised learning of contextual role knowledge for coreference resolution. In *Proceedings of the Human Language Technology Conference / North American Chapter of the Association for Computational Linguistics Annual Meeting (HLT/NAACL-04)*.
- Bryl, V.; Giuliano, C.; Serafini, L.; and Tymoshenko, K. 2010a. Supporting natural language processing with background knowledge: coreference resolution case. In *Proceedings of the 9th International Semantic Web Conference (ISWC2010)*.
- Bryl, V.; Giuliano, C.; Serafini, L.; and Tymoshenko, K. 2010b. Using background knowledge to support coreference resolution. In *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI 2010)*.
- Garera, N., and Yarowsky, D. 2006. Resolving and generating definite anaphora by modeling hypernymy using unlabeled corpora. In *Proceedings of the Conference on Natural Language Learning*.
- Giuliano, C.; Gliozzo, A. M.; and Strapparava, C. 2009. Kernel methods for minimally supervised WSD. *Computational Linguistics* 35(4):513–528.
- Haghighi, A., and Klein, D. 2009. Simple coreference resolution with rich syntactic and semantic features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, 1152–1161.
- Harabagiu, S.; Bunescu, R.; and Maiorano, S. 2001. Text and knowledge mining for coreference resolution. In *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics*, 55–62.
- Huang, Z. H.; G., Z.; W., X.; and A., C. 2009. Effectively exploiting WordNet in semantic class classification for coreference resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.
- McCallum, A., and Wellner, B. 2003. Toward conditional models of identity uncertainty with application to proper noun coreference. In *Proceedings of the IJCAI Workshop on Information Integration on the Web*.
- Poesio, M.; Uryupina, O.; Vieira, R.; Alexandrov-Kabadjov, M.; and Goulart, R. 2004. Discourse-new detectors for definite description resolution: a survey and preliminary proposal. In *Proceedings of the Reference Resolution Workshop at ACL'04*.
- Ponzetto, S. P., and Poesio, M. 2009. State-of-the-art NLP approaches to coreference resolution: theory and practical recipes, ACL-09 tutorial.
- Ponzetto, S. P., and Strube, M. 2006. Exploiting semantic role labeling, WordNet and Wikipedia for coreference resolution. In *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, 192–199.
- Poon, H., and Domingos, P. 2008. Joint unsupervised coreference resolution with Markov logic. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.
- Shawe-Taylor, J., and Cristianini, N. 2004. *Kernel Methods for Pattern Analysis*. Cambridge University Press.
- Soon, W. M.; Ng, H. T.; and Lim, D. C. Y. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics* 27(4):521–544.
- Suchanek, F. M.; Kasneci, G.; and Weikum, G. 2007. Yago: a core of semantic knowledge. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, 697–706. ACM Press.
- Uryupina, O. 2004. Evaluating name-matching for coreference resolution. In *Proceedings of the Language Resources and Evaluation Conference*.
- Versley, Y.; Moschitti, A.; Poesio, M.; and Yang, X. 2008a. Coreference systems based on kernel methods. In *Proceedings of the 22nd International Conference on Computational Linguistics*.
- Versley, Y.; Ponzetto, S. P.; Poesio, M.; Eidelman, V.; Jern, A.; Smith, J.; Yang, X.; and Moschitti, A. 2008b. BART: a modular toolkit for coreference resolution. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies*, 9–12.
- Yang, X., and Su, J. 2007. Coreference resolution using semantic relatedness information from automatically discovered patterns. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 528–535.