First-Order Probabilistic Models for Coreference Resolution

Aron Culotta and **Michael Wick** and **Robert Hall** and **Andrew McCallum**
Department of Computer Science
University of Massachusetts
Amherst, MA 01003
{culotta,mwick,rhall,mccallum}@cs.umass.edu

Abstract

Traditional noun phrase coreference resolution systems represent features only of pairs of noun phrases. In this paper, we propose a machine learning method that enables features over *sets* of noun phrases, resulting in a first-order probabilistic model for coreference. We outline a set of approximations that make this approach practical, and apply our method to the ACE coreference dataset, achieving a 45% error reduction over a comparable method that only considers features of pairs of noun phrases. This result demonstrates an example of how a first-order logic representation can be incorporated into a probabilistic model and scaled efficiently.

1 Introduction

Noun phrase coreference resolution is the problem of clustering noun phrases into anaphoric sets. A standard machine learning approach is to perform a set of independent binary classifications of the form "*Is mention a coreferent with mention b?*"

This approach of decomposing the problem into pairwise decisions presents at least two related difficulties. First, it is not clear how best to convert the set of pairwise classifications into a disjoint clustering of noun phrases. The problem stems from the transitivity constraints of coreference: If a and b are coreferent, and b and c are coreferent, then a and c must be coreferent.

This problem has recently been addressed by a number of researchers. A simple approach is to perform the transitive closure of the pairwise decisions. However, as shown in recent work (McCallum and Wellner, 2003; Singla and Domingos, 2005), better performance can be obtained by performing *relational inference* to directly consider the dependence among a set of predictions. For example, McCallum and Wellner (2005) apply a graph partitioning algorithm on a weighted, undirected graph in which vertices are noun phrases and edges are weighted by the pairwise score between noun phrases.

A second and less studied difficulty is that the pairwise decomposition restricts the feature set to evidence about *pairs* of noun phrases only. This restriction can be detrimental if there exist features of *sets* of noun phrases that cannot be captured by a combination of pairwise features. As a simple example, consider prohibiting coreferent sets that consist only of pronouns. That is, we would like to require that there be at least one antecedent for a set of pronouns. The pairwise decomposition does not make it possible to capture this constraint.

In general, we would like to construct arbitrary features over a cluster of noun phrases using the full expressivity of first-order logic. Enabling this sort of flexible representation within a statistical model has been the subject of a long line of research on *first-order probabilistic models* (Gaifman, 1964; Halpern, 1990; Paskin, 2002; Poole, 2003; Richardson and Domingos, 2006).

Conceptually, a first-order probabilistic model can be described quite compactly. A configuration of the world is represented by a set of predi-

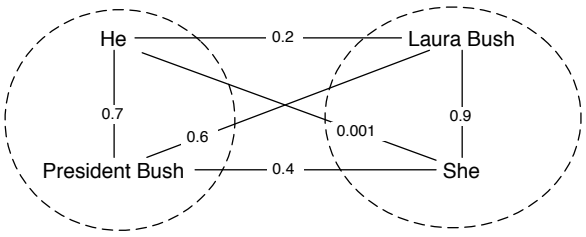

Figure 1: An example noun coreference graph in which vertices are noun phrases and edge weights are proportional to the probability that the two nouns are coreferent. Partitioning such a graph into disjoint clusters corresponds to performing coreference resolution on the noun phrases.

cates, each of which has an associated real-valued parameter. The likelihood of each configuration of the world is proportional to a combination of these weighted predicates. In practice, however, enumerating all possible configurations, or even all the predicates of one configuration, can result in intractable combinatorial growth (de Salvo Braz et al., 2005; Culotta and McCallum, 2006).

In this paper, we present a practical method to perform training and inference in first-order models of coreference. We empirically validate our approach on the ACE coreference dataset, showing that the first-order features can lead to an 45% error reduction.

2 Pairwise Model

In this section we briefly review the standard pairwise coreference model. Given a pair of noun phrases $x_{ij} = \{x_i, x_j\}$, let the binary random variable y_{ij} be 1 if x_i and x_j are coreferent. Let $F = \{f_k(x_{ij}, y)\}$ be a set of features over x_{ij}. For example, $f_k(x_{ij}, y)$ may indicate whether x_i and x_j have the same gender or number. Each feature f_k has an associated real-valued parameter λ_k. The pairwise model is

$$p(y_{ij}|x_{ij}) = \frac{1}{Z_{x_{ij}}} \exp \sum_k \lambda_k f_k(x_{ij}, y_{ij})$$

where $Z_{x_{ij}}$ is a normalizer that sums over the two settings of y_{ij}.

This is a maximum-entropy classifier (i.e. logistic regression) in which $p(y_{ij}|x_{ij})$ is the probability that x_i and x_j are coreferent. To estimate $\Lambda = \{\lambda_k\}$ from labeled training data, we perform gradient ascent to maximize the log-likelihood of the labeled data.

Two critical decisions for this method are (1) how to sample the training data, and (2) how to combine the pairwise predictions at test time. Systems often perform better when these decisions complement each other.

Given a data set in which noun phrases have been manually clustered, the training data can be created by simply enumerating over each pair of noun phrases x_{ij}, where y_{ij} is true if x_i and x_j are in the same cluster. However, this approach generates a highly unbalanced training set, with negative examples outnumbering positive examples. Instead, Soon et al. (2001) propose the following sampling method: Scan the document from left to right. Compare each noun phrase x_i to each preceding noun phrase x_j, scanning from right to left. For each pair x_i, x_j, create a training instance $\langle x_{ij}, y_{ij} \rangle$, where y_{ij} is 1 if x_i and x_j are coreferent. The scan for x_j terminates when a positive example is constructed, or the beginning of the document is reached. This results in a training set that has been pruned of distant noun phrase pairs.

At testing time, we can construct an undirected, weighted graph in which vertices correspond to noun phrases and edge weights are proportional to $p(y_{ij}|x_{ij})$. The problem is then to partition the graph into clusters with high *intra-cluster* edge weights and low *inter-cluster* edge weights. An example of such a graph is shown in Figure 1.

Any partitioning method is applicable here; however, perhaps most common for coreference is to perform greedy clustering guided by the word order of the document to complement the sampling method described above (Soon et al., 2001). More precisely, scan the document from left-to-right, assigning each noun phrase x_i to the same cluster as the closest *preceding* noun phrase x_j for which $p(y_{ij}|x_{ij}) > \delta$, where δ is some classification threshold (typically 0.5). Note that this method contrasts with standard greedy agglomerative clustering, in which each noun phrase would be assigned to the *most probable* cluster according to $p(y_{ij}|x_{ij})$.

Choosing the closest preceding phrase is common because nearby phrases are a priori more likely to be coreferent.

We refer to the training and inference methods described in this section as the Pairwise Model.

3 First-Order Logic Model

We propose augmenting the Pairwise Model to enable classification decisions over *sets* of noun phrases.

Given a set of noun phrases $\mathbf{x}^j = \{x_i\}$, let the binary random variable y_j be 1 if *all* the noun phrases $x_i \in \mathbf{x}^j$ are coreferent. The features f_k and weights λ_k are defined as before, but now the features can represent arbitrary attributes over the entire set $\mathbf{x}^j$. This allows us to use the full flexibility of first-order logic to construct features about sets of nouns. The First-Order Logic Model is

$$p(y_j|\mathbf{x}^j) = \frac{1}{Z_{\mathbf{x}^j}} \exp \sum_k \lambda_k f_k(\mathbf{x}^j, y_j)$$

where $Z_{\mathbf{x}^j}$ is a normalizer that sums over the two settings of y_j.

Note that this model gives us the representational power of recently proposed Markov logic networks (Richardson and Domingos, 2006); that is, we can construct arbitrary formulae in first-order logic to characterize the noun coreference task, and can learn weights for instantiations of these formulae. However, naively *grounding* the corresponding Markov logic network results in a combinatorial explosion of variables. Below we outline methods to scale training and prediction with this representation.

As in the Pairwise Model, we must decide how to sample training examples and how to combine independent classifications at testing time. It is important to note that by moving to the First-Order Logic Model, the number of possible predictions has increased exponentially. In the Pairwise Model, the number of possible y variables is $O(|\mathbf{x}|^2)$, where $\mathbf{x}$ is the set of noun phrases. In the First-Order Logic Model, the number of possible y variables is $O(2^{|\mathbf{x}|})$: There is a y variable for each possible element of the powerset of $\mathbf{x}$. Of course, we do not enumerate this set; rather, we incrementally instantiate y variables as needed during prediction.

A simple method to generate training examples is to sample positive and negative cluster examples uniformly at random from the training data. Positive examples are generated by first sampling a true cluster, then sampling a subset of that cluster. Negative examples are generated by sampling two positive examples and merging them into the same cluster.

At testing time, we perform standard greedy agglomerative clustering, where the score for each merger is proportional to the probability of the newly formed clustering according to the model. Clustering terminates when there exists no additional merge that improves the probability of the clustering.

We refer to the system described in this section as First-Order Uniform.

4 Error-driven and Rank-based training of the First-Order Model

In this section we propose two enhancements to the training procedure for the First-Order Uniform model.

First, because each training example consists of a subset of noun phrases, the number of possible training examples we can generate is exponential in the number of noun phrases. We propose an error-driven sampling method that generates training examples from errors the model makes on the training data. The algorithm is as follows: Given initial parameters Λ, perform greedy agglomerative clustering on training document i until an incorrect cluster is formed. Update the parameter vector according to this mistake, then repeat for the next training document. This process is repeated for a fixed number of iterations.

Exactly how to update the parameter vector is addressed by the second enhancement. We propose modifying the optimization criterion of training to perform *ranking* rather than *classification* of clusters. Consider a training example cluster with a negative label, indicating that not all of the noun phrases it contains are coreferent. A classification training algorithm will "penalize" all the features associated with this cluster, since they correspond to a negative example. However, because there may exists subsets of the cluster that *are* coreferent, features representing these positive subsets may be unjustly penalized.

To address this problem, we propose constructing training examples consisting of one negative exam-

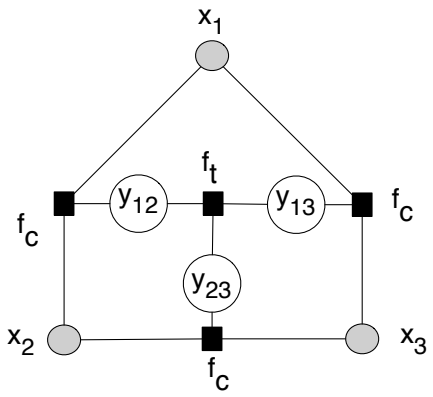

Figure 2: An example noun coreference factor graph for the Pairwise Model in which factors f_c model the coreference between two nouns, and f_t enforce the transitivity among related decisions. The number of y variables increases quadratically in the number of x variables.

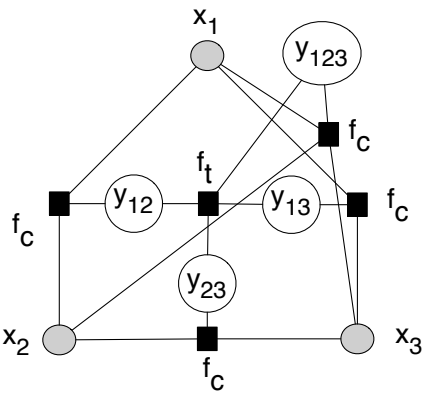

Figure 3: An example noun coreference factor graph for the First-Order Model in which factors f_c model the coreference between *sets* of nouns, and f_t enforce the transitivity among related decisions. Here, the additional node y_{123} indicates whether nouns $\{x_1, x_2, x_3\}$ are all coreferent. The number of y variables increases exponentially in the number of x variables.

ple and one "nearby" positive example. In particular, when agglomerative clustering incorrectly merges two clusters, we select the resulting cluster as the negative example, and select as the positive example a cluster that can be created by merging other existing clusters.[1] We then update the weight vector so that the positive example is assigned a higher score than the negative example. This approach allows the update to only penalize the *difference* between the two features of examples, thereby not penalizing features representing any overlapping coreferent clusters.

To implement this update, we use MIRA (Margin Infused Relaxed Algorithm), a relaxed, online maximum margin training algorithm (Crammer and Singer, 2003). It updates the parameter vector with two constraints: (1) the positive example must have a higher score by a given margin, and (2) the change to Λ should be minimal. This second constraint is to reduce fluctuations in Λ. Let $s^+(\Lambda, \mathbf{x}^j)$ be the unnormalized score for the positive example and $s^-(\Lambda, \mathbf{x}^k)$ be the unnormalized score of the negative example. Each update solves the following quadratic program:

$$\Lambda^{t+1} = \underset{\Lambda}{\operatorname{argmin}} ||\Lambda^t - \Lambda||^2$$
$$\text{s.t.}$$
$$s^+(\Lambda, \mathbf{x}^j) - s^-(\Lambda, \mathbf{x}^k) \geq 1$$

In this case, MIRA with a single constraint can be efficiently solved in one iteration of the Hildreth and D'Esopo method (Censor and Zenios, 1997). Additionally, we average the parameters calculated at each iteration to improve convergence.

We refer to the system described in this section as First-Order MIRA.

5 Probabilistic Interpretation

In this section, we describe the Pairwise and First-Order models in terms of the factor graphs they approximate.

For the Pairwise Model, a corresponding undirected graphical model can be defined as

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_\mathbf{x}} \prod_{y_{ij} \in \mathbf{y}} f_c(y_{ij}, x_{ij}) \prod_{y_{ij}, y_{jk} \in \mathbf{y}} f_t(y_{ij}, y_{j,k}, y_{ik}, x_{ij}, x_{jk}, x_{ik})$$

[1] Of the possible positive examples, we choose the one with the highest probability under the current model to guard against large fluctuations in parameter updates

where $Z_{\mathbf{x}}$ is the input-dependent normalizer and factor f_c parameterizes the pairwise noun phrase compatibility as $f_c(y_{ij}, x_{ij}) = \exp(\sum_k \lambda_k f_k(y_{ij}, x_{ij}))$. Factor f_t enforces the transitivity constraints by $f_t(\cdot) = -\infty$ if transitivity is not satisfied, 1 otherwise. This is similar to the model presented in McCallum and Wellner (2005). A factor graph for the Pairwise Model is presented in Figure 2 for three noun phrases.

For the First-Order model, an undirected graphical model can be defined as

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_{\mathbf{x}}} \prod_{y_j \in \mathbf{y}} f_c(y_j, \mathbf{x}^j) \prod_{y_j \in \mathbf{y}} f_t(y_j, \mathbf{x}^j)$$

where $Z_{\mathbf{x}}$ is the input-dependent normalizer and factor f_c parameterizes the cluster-wise noun phrase compatibility as $f_c(y_j, \mathbf{x}^j) = \exp(\sum_k \lambda_k f_k(y_j, x^j))$. Again, factor f_t enforces the transitivity constraints by $f_t(\cdot) = -\infty$ if transitivity is not satisfied, 1 otherwise. Here, transitivity is a bit more complicated, since it also requires that if $y_j = 1$, then for any subset $\mathbf{x}^k \subseteq \mathbf{x}^j$, $y_k = 1$. A factor graph for the First-Order Model is presented in Figure 3 for three noun phrases.

The methods described in Sections 2, 3 and 4 can be viewed as estimating the parameters of each factor f_c independently. This approach can therefore be viewed as a type of *piecewise approximation* of exact parameter estimation in these models (Sutton and McCallum, 2005). Here, each f_c is a "piece" of the model trained independently. These pieces are combined at prediction time using clustering algorithms to enforce transitivity. Sutton and McCallum (2005) show that such a piecewise approximation can be theoretically justified as minimizing an upper bound of the exact loss function.

6 Experiments

6.1 Data

We apply our approach to the noun coreference ACE 2004 data, containing 443 news documents with 28,135 noun phrases to be coreferenced. 336 documents are used for training, and the remainder for testing. All entity types are candidates for coreference (pronouns, named entities, and nominal entities). We use the true entity segmentation, and parse each sentence in the corpus using a phrase-structure grammar, as is common for this task.

6.2 Features

We follow Soon et al. (2001) and Ng and Cardie (2002) to generate most of our features for the Pairwise Model. These include:

- Match features - Check whether gender, number, head text, or entire phrase matches
- Mention type (pronoun, name, nominal)
- Aliases - Heuristically decide if one noun is the acronym of the other
- Apposition - Heuristically decide if one noun is in apposition to the other
- Relative Pronoun - Heuristically decide if one noun is a relative pronoun referring to the other.
- Wordnet features - Use Wordnet to decide if one noun is a hypernym, synonym, or antonym of another, or if they share a hypernym.
- Both speak - True if both contain an adjacent context word that is a synonym of "said." This is a domain-specific feature that helps for many newswire articles.
- Modifiers Match - for example, in the phrase "President Clinton", "President" is a modifier of "Clinton". This feature indicates if one noun is a modifier of the other, or they share a modifier.
- Substring - True if one noun is a substring of the other (e.g. "Egypt" and "Egyptian").

The First-Order Model includes the following features:

- Enumerate each pair of noun phrases and compute the features listed above. **All-X** is true if all pairs share a feature X, **Most-True-X** is true if the majority of pairs share a feature X, and **Most-False-X** is true if most of the pairs do not share feature X.

- Use the output of the Pairwise Model for each pair of nouns. **All-True** is true if all pairs are predicted to be coreferent, **Most-True** is true if most pairs are predicted to be coreferent, and **Most-False** is true if most pairs are predicted to not be coreferent. Additionally, **Max-True** is true if the maximum pairwise score is above threshold, and **Min-True** if the minimum pairwise score is above threshold.

- Cluster Size indicates the size of the cluster.

- Count how many phrases in the cluster are of each mention type (name, pronoun, nominal), number (singular/plural) and gender (male/female). The features **All-X** and **Most-True-X** indicate how frequent each feature is in the cluster. This feature can capture the soft constraint such that no cluster consists only of pronouns.

In addition to the listed features, we also include conjunctions of size 2, for example "Genders match AND numbers match".

6.3 Evaluation

We use the B^3 algorithm to evaluate the predicted coreferent clusters (Amit and Baldwin, 1998). B^3 is common in coreference evaluation and is similar to the precision and recall of coreferent links, except that systems are rewarded for singleton clusters. For each noun phrase x_i, let c_i be the number of mentions in x_i's predicted cluster that are in fact coreferent with x_i (including x_i itself). Precision for x_i is defined as c_i divided by the number of noun phrases in x_i's cluster. Recall for x_i is defined as the c_i divided by the number of mentions in the gold standard cluster for x_i. $F1$ is the harmonic mean of recall and precision.

6.4 Results

In addition to Pairwise, First-Order Uniform, and First-Order MIRA, we also compare against Pairwise MIRA, which differs from First-Order MIRA only by the fact that it is restricted to pairwise features.

Table 1 suggests both that first-order features and error-driven training can greatly improve performance. The First-Order Model outperforms the Pairwise Model in F1 measure for both standard training and error-driven training. We attribute some of this improvement to the capability of the First-Order model to capture features of entire clusters that may indicate some phrases are not coreferent. Also, we attribute the gains from error-driven training to the fact that training examples are generated based on errors made on the training data. (However, we should note that there are also small differences in the feature sets used for error-driven and standard training results.)

	F1	**Prec**	**Rec**
First-Order MIRA	**79.3**	86.7	73.2
Pairwise MIRA	72.5	92.0	59.8
First-Order Uniform	**69.2**	79.0	61.5
Pairwise	62.4	62.5	62.3

Table 1: B^3 results for ACE noun phrase coreference. FIRST-ORDER MIRA is our proposed model that takes advantage of first-order features of the data and is trained with error-driven and rank-based methods. We see that both the first-order features and the training enhancements improve performance consistently.

Error analysis indicates that often noun x_i is correctly not merged with a cluster $\mathbf{x}^j$ when $\mathbf{x}^j$ has a strong internal coherence. For example, if all 5 mentions of *France* in a document are string identical, then the system will be extremely cautious of merging a noun that is not equivalent to *France* into $\mathbf{x}^j$, since this will turn off the "All-String-Match" feature for cluster $\mathbf{x}^j$.

To our knowledge, the best results on this dataset were obtained by the meta-classification scheme of Ng (2005). Although our train-test splits may differ slightly, the best B-Cubed F1 score reported in Ng (2005) is 69.3%, which is considerably lower than the 79.3% obtained with our method. Also note that the Pairwise baseline obtains results similar to those in Ng and Cardie (2002).

7 Related Work

There has been a recent interest in training methods that enable the use of first-order features (Paskin, 2002; Daumé III and Marcu, 2005b; Richardson and Domingos, 2006). Perhaps the most related is

"learning as search optimization" (LASO) (Daumé III and Marcu, 2005b; Daumé III and Marcu, 2005a). Like the current paper, LASO is also an error-driven training method that integrates prediction and training. However, whereas we explicitly use a ranking-based loss function, LASO uses a binary classification loss function that labels each candidate structure as *correct* or *incorrect*. Thus, each LASO training example contains *all* candidate predictions, whereas our training examples contain only the highest scoring incorrect prediction and the highest scoring correct prediction. Our experiments show the advantages of this ranking-based loss function. Additionally, we provide an empirical study to quantify the effects of different example generation and loss function decisions.

Collins and Roark (2004) present an incremental perceptron algorithm for parsing that uses "early update" to update the parameters when an error is encountered. Our method uses a similar "early update" in that training examples are only generated for the *first* mistake made during prediction. However, they do not investigate rank-based loss functions.

Others have attempted to train global scoring functions using Gibbs sampling (Finkel et al., 2005), message propagation, (Bunescu and Mooney, 2004; Sutton and McCallum, 2004), and integer linear programming (Roth and Yih, 2004). The main distinctions of our approach are that it is simple to implement, not computationally intensive, and adaptable to arbitrary loss functions.

There have been a number of machine learning approaches to coreference resolution, traditionally factored into classification decisions over pairs of nouns (Soon et al., 2001; Ng and Cardie, 2002). Nicolae and Nicolae (2006) combine pairwise classification with graph-cut algorithms. Luo et al. (2004) do enable features between mention-cluster pairs, but do not perform the error-driven and ranking enhancements proposed in our work. Denis and Baldridge (2007) use a ranking loss function for pronoun coreference; however the examples are still pairs of pronouns, and the example generation is not error driven. Ng (2005) learns a meta-classifier to choose the best prediction from the output of several coreference systems. While in theory a meta-classifier can flexibly represent features, they do not explore features using the full flexibility of first-order logic. Also, their method is neither error-driven nor rank-based.

McCallum and Wellner (2003) use a conditional random field that factors into a product of pairwise decisions about pairs of nouns. These pairwise decisions are made collectively using relational inference; however, as pointed out in Milch et al. (2004), this model has limited representational power since it does not capture features of *entities*, only of pairs of mention. Milch et al. (2005) address these issues by constructing a generative probabilistic model, where noun clusters are sampled from a generative process. Our current work has similar representational flexibility as Milch et al. (2005) but is discriminatively trained.

8 Conclusions and Future Work

We have presented learning and inference procedures for coreference models using first-order features. By relying on sampling methods at training time and approximate inference methods at testing time, this approach can be made scalable. This results in a coreference model that can capture features over *sets* of noun phrases, rather than simply *pairs* of noun phrases.

This is an example of a model with extremely flexible representational power, but for which exact inference is intractable. The simple approximations we have described here have enabled this more flexible model to outperform a model that is simplified for tractability.

A short-term extension would be to consider features over entire *clusterings*, such as the number of clusters. This could be incorporated in a ranking scheme, as in Ng (2005).

Future work will extend our approach to a wider variety of tasks. The model we have described here is specific to clustering tasks; however a similar formulation could be used to approach a number of language processing tasks, such as parsing and relation extraction. These tasks could benefit from first-order features, and the present work can guide the approximations required in those domains.

Additionally, we are investigating more sophisticated inference algorithms that will reduce the greediness of the search procedures described here.

Acknowledgments

This work was supported in part by the Defense Advanced Research Projects Agency (DARPA), through the Department of the Interior, NBC, Acquisition Services Division, under contract #NBCHD030010, in part by U.S. Government contract #NBCH040171 through a subcontract with BBNT Solutions LLC, in part by The Central Intelligence Agency, the National Security Agency and National Science Foundation under NSF grant #IIS-0326249, in part by Microsoft Live Labs, and in part by the Defense Advanced Research Projects Agency (DARPA) under contract #HR0011-06-C-0023. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s)' and do not necessarily reflect those of the sponsor.

References

B. Amit and B. Baldwin. 1998. Algorithms for scoring coreference chains. In *Proceedings of the Seventh Message Understanding Conference (MUC7)*.

Razvan Bunescu and Raymond J. Mooney. 2004. Collective information extraction with relational markov networks. In *ACL*.

Y. Censor and S.A. Zenios. 1997. *Parallel optimization : theory, algorithms, and applications*. Oxford University Press.

Michael Collins and Brian Roark. 2004. Incremental parsing with the perceptron algorithm. In *ACL*.

Koby Crammer and Yoram Singer. 2003. Ultraconservative online algorithms for multiclass problems. *JMLR*, 3:951–991.

Aron Culotta and Andrew McCallum. 2006. Tractable learning and inference with high-order representations. In *ICML Workshop on Open Problems in Statistical Relational Learning*, Pittsburgh, PA.

Hal Daumé III and Daniel Marcu. 2005a. A large-scale exploration of effective global features for a joint entity detection and tracking model. In *HLT/EMNLP*, Vancouver, Canada.

Hal Daumé III and Daniel Marcu. 2005b. Learning as search optimization: Approximate large margin methods for structured prediction. In *ICML*, Bonn, Germany.

Rodrigo de Salvo Braz, Eyal Amir, and Dan Roth. 2005. Lifted first-order probabilistic inference. In *IJCAI*, pages 1319–1325.

Pascal Denis and Jason Baldridge. 2007. A ranking approach to pronoun resolution. In *IJCAI*.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *ACL*, pages 363–370.

H. Gaifman. 1964. Concerning measures in first order calculi. *Israel J. Math*, 2:1–18.

J. Y. Halpern. 1990. An analysis of first-order logics of probability. *Artificial Intelligence*, 46:311–350.

Xiaoqiang Luo, Abe Ittycheriah, Hongyan Jing, Nanda Kambhatla, and Salim Roukos. 2004. A mention-synchronous coreference resolution algorithm based on the Bell tree. In *ACL*, page 135.

A. McCallum and B. Wellner. 2003. Toward conditional models of identity uncertainty with application to proper noun coreference. In *IJCAI Workshop on Information Integration on the Web*.

Andrew McCallum and Ben Wellner. 2005. Conditional models of identity uncertainty with application to noun coreference. In Lawrence K. Saul, Yair Weiss, and Léon Bottou, editors, *NIPS17*. MIT Press, Cambridge, MA.

Brian Milch, Bhaskara Marthi, and Stuart Russell. 2004. BLOG: Relational modeling with unknown objects. In *ICML 2004 Workshop on Statistical Relational Learning and Its Connections to Other Fields*.

Brian Milch, Bhaskara Marthi, Stuart Russell, David Sontag, Daniel L. Ong, and Andrey Kolobov. 2005. BLOG: Probabilistic models with unknown objects. In *IJCAI*.

Vincent Ng and Claire Cardie. 2002. Improving machine learning approaches to coreference resolution. In *ACL*.

Vincent Ng. 2005. Machine learning for coreference resolution: From local classification to global ranking. In *ACL*.

Cristina Nicolae and Gabriel Nicolae. 2006. Bestcut: A graph algorithm for coreference resolution. In *EMNLP*, pages 275–283, Sydney, Australia, July. Association for Computational Linguistics.

Mark A. Paskin. 2002. Maximum entropy probabilistic logic. Technical Report UCB/CSD-01-1161, University of California, Berkeley.

D. Poole. 2003. First-order probabilistic inference. In *IJCAI*, pages 985–991, Acapulco, Mexico. Morgan Kaufman.

Matthew Richardson and Pedro Domingos. 2006. Markov logic networks. *Machine Learning*, 62:107–136.

D. Roth and W. Yih. 2004. A linear programming formulation for global inference in natural language tasks. In *The 8th Conference on Compuational Natural Language Learning*, May.

Parag Singla and Pedro Domingos. 2005. Discriminative training of markov logic networks. In *AAAI*, Pittsburgh, PA.

Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. 2001. A machine learning approach to coreference resolution of noun phrases. *Comput. Linguist.*, 27(4):521–544.

Charles Sutton and Andrew McCallum. 2004. Collective segmentation and labeling of distant entities in information extraction. Technical Report TR # 04-49, University of Massachusetts, July.

Charles Sutton and Andrew McCallum. 2005. Piecewise training of undirected models. In *21st Conference on Uncertainty in Artificial Intelligence*.