A Constrained Latent Variable Model for Coreference Resolution

Kai-Wei Chang Rajhans Samdani Dan Roth
University of Illinois at Urbana-Champaign
{kchang10|rsamdan2|danr}@illinois.edu

Abstract

Coreference resolution is a well known clustering task in Natural Language Processing. In this paper, we describe the Latent Left Linking model (L^3M), a novel, principled, and linguistically motivated latent structured prediction approach to coreference resolution. We show that L^3M admits efficient inference and can be augmented with knowledge-based constraints; we also present a fast stochastic gradient based learning. Experiments on ACE and Ontonotes data show that L^3M and its constrained version, CL^3M, are more accurate than several state-of-the-art approaches as well as some structured prediction models proposed in the literature.

1 Introduction

Coreference resolution is a challenging task, that involves identification and clustering of noun phrases mentions that refer to the same real-world entity. Most machine learning approaches to coreference resolution learn a scoring function to estimate the compatibility between two mentions or two sets of previously clustered mentions. Then, a decoding algorithm is designed to aggregate these scores and find an optimal clustering assignment.

The most popular of these frameworks is the pairwise mention model (Soon et al., 2001; Ng and Cardie, 2002; Bengtson and Roth, 2008), which learns a compatibility score of mention-pairs and uses these pairwise scores to obtain a global clustering. Recently, efforts have been made (Haghighi and Klein, 2010; Rahman and Ng, 2011b; Rahman and Ng, 2011c) to consider models that capture higher order interactions, in particular, between mentions and previously identified entities (that is, between mentions and clusters). While such models are potentially more expressive, they are largely based on heuristics to achieve computational tractability.

This paper focuses on a novel and principled machine learning framework that pushes the state-of-the-art while operating at a mention-pair granularity. We present two models — the Latent Left-Linking Model (L^3M), and a version of that is augmented with domain knowledge-based constraints, the Constrained Latent Left-Linking Model (CL^3M). L^3M admits efficient inference, linking each mention to a previously occurring mention to its left, much like the existing best-left-link inference models (Ng and Cardie, 2002; Bengtson and Roth, 2008). However, unlike previous best-link techniques, learning in our case is performed jointly with decoding — we present a novel latent structural SVM approach, optimized using a fast stochastic gradient-based technique. Furthermore, we present a probabilistic generalization of L^3M that is more expressive in that it is capable of considering mention-entity interactions using scores at the mention-pair granularity. We augment this model with a temperature-like parameter (Samdani et al., 2012) to provide additional flexibility.

CL^3M augments L^3M with knowledge-based constraints following (Roth and Yih, 2004; Denis and Baldridge, 2007). This capability is very desirable as shown by the success of the rule-based deterministic approach of Raghunathan et al. (2010) in the CoNLL shared task 2011 (Pradhan et al., 2011). In L^3M, domain-specific constraints are incorporated into learning and inference in a straightforward way. CL^3M scores a mention's contribution to its cluster by combining the corresponding score

Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 601–612,
Seattle, Washington, USA, 18-21 October 2013. ©2013 Association for Computational Linguistics

of the underlying L^3M model with that from a set of constraints.

Most importantly, in our experiments on benchmark coreference datasets, we show that CL^3M, with just five constraints, compares favorably with other, more complicated, state-of-the-art algorithms on a variety of evaluation metrics. Overall, the main contribution of this paper is a principled machine learning model operating at mention-pair granularity, using easy to implement constraint-augmented inference and learning, that yields competitive results on coreference resolution on Ontonotes-5.0 (Pradhan et al., 2012) and ACE 2004 (NIST, 2004).

2 Related Work

The idea of Latent Left-linking Model (L^3M) is inspired by a popular inference approach to coreference which we call the *Best-Left-Link* approach (Ng and Cardie, 2002; Bengtson and Roth, 2008). In the best-left-link strategy, each mention i is connected to the best antecedent mention j with $j < i$ (i.e. a mention occurring to the left of i, assuming a left-to-right reading order), thereby creating a *left-link*. The "best" antecedent mention is the one with the highest pairwise score, w_{ij}; furthermore, if w_{ij} is below some threshold, say 0, then i is not connected to any antecedent mention. The final clustering is a transitive closure of these "best" links. The intuition behind best-left-link strategy is based on how humans read and decipher coreference links – they mostly rely on information to the left of the mention when deciding whether to add it to a previously constructed cluster or not. This strategy has been successful and commonly used in coreference resolution (Ng and Cardie, 2002; Bengtson and Roth, 2008; Stoyanov et al., 2009). However, most works have developed ad-hoc approaches to implement this idea. For instance, Bengtson and Roth (2008) train a model $\mathbf{w}$ on binary training data generated by taking for each mention, the closest antecedent coreferent mention as a positive example, and all the other mentions as negative examples. Similar approaches to training and, additionally, decoupling the training stage from the clustering stage were used by other systems. In this paper, we formalize the learning problem of the best-left-link model as a structured prediction problem and analyze our system with detailed experiments. Furthermore, we generalize this approach by considering multiple pairwise left-links instead of just the best link, efficiently capturing the notion of a mention-to-cluster link.

Many techniques in the coreference literature break away from the mention pair-based, best-left-link paradigm. Denis and Baldridge (2008) and Ng (2005) learn a local ranker to rank the mention pairs based on their compatibility. While these approaches achieve decent empirical performance, it is unclear why these are the right ways to train the model. Some techniques consider a more expressive model by using features defined over mention-cluster or cluster-cluster (Rahman and Ng, 2011c; Stoyanov and Eisner, 2012; Haghighi and Klein, 2010). For these models, the inference and learning algorithms are usually complicated. Very recently, Durrett et al. (2013) propose a probabilistic model which enforces structural agreement constraints between specified properties of mention cluster when using a mention-pair model. This approach is very related to the probabilistic extension of our method as both models attempt to leverage entity-level information from mention-pair features. However, our approach is simpler because it directly considers the probabilities of multiple links. Furthermore, while their model performs only slightly better than the Stanford rule-based system (Lee et al., 2011), we significantly outperform this system. Most importantly, our model obtains state-of-the-art performance on OntoNotes-5.0 while still operating at the mention-pair granularity. We believe that this is due to our novel and principled structured prediction framework which results in accurate (and efficient) training.

Several structured prediction techniques have been applied to coreference resolution in the machine learning literature. For example, McCallum and Wellner (2003) and Finley and Joachims (2005) model coreference as a correlational clustering problem (Bansal et al., 2002) on a complete graph over the mentions with edge weights given by the pairwise classifier. However, correlational clustering is known to be NP Hard (Bansal et al., 2002); nonetheless, an ILP solver or an approximate inference algorithm can be used to solve this problem. Another approach proposed by Yu and Joachims (2009) formu-

lates coreference with latent spanning trees. However, their approach has no directionality between mentions, whereas our latent structure captures the natural left-to-right ordering of mentions. In our experiments (Sec. 5), we show that our technique vastly outperforms both the spanning tree and the correlational clustering techniques. We also compare with (Fernandes et al., 2012) and the publicly available Stanford coreference system (Raghunathan et al., 2010; Lee et al., 2011), a state-of-the-art rule-based system.

Finally, some research (Ratinov and Roth, 2012; Bansal and Klein, 2012; Rahman and Ng, 2011a) has tried to integrate world knowledge from web-based statistics or knowledge bases into a coreference system. World knowledge is potentially useful for resolving coreference and can be injected into our system in a straightforward way via the constraints framework. We will show an example of incorporating our system with name-entity and WordNet-based similarity metric (Q. Do, 2009) in Sec. 5. Including massive amount of information from knowledge resources is not the focus of this paper and may distort the comparison with other relevant models but our results indicate that this is doable in our model, and may provide significant improvements.

3 Latent Left-Linking Model with Constraints

In this section, we describe our Constrained Latent Left-Linking Model (CL^{3}M). CL^{3}M is inspired by a few ideas from the literature: (a) the popular *Best-Left-Link* inference approach to coreference (Ng and Cardie, 2002; Bengtson and Roth, 2008), and (b) the injection of domain knowledge-based constraints for structured prediction (Roth and Yih, 2004; Clarke and Lapata, 2006; Chang et al., 2012b; Ganchev et al., 2010; Koo et al., 2010; Pascal and Baldridge, 2009).

We first introduce the notion of a pairwise mention-scorer, then introduce our Left-Linking Model (L^{3}M), and finally describe how to inject constraints into our model.

Let d be a document with m_d mentions. Mentions are denoted solely using their indices, ranging from 1 to m_d. A coreference clustering $\mathcal{C}$ for document d is a collection of disjoint sets partitioning the set $\{1, \ldots, m_d\}$. We represent $\mathcal{C}$ as a binary function with $\mathcal{C}(i, j) = 1$ if mentions i and j are coreferent, otherwise $\mathcal{C}(i, j) = 0$. Let $s(\mathcal{C}; \mathbf{w}, d)$ be the score of a given clustering $\mathcal{C}$ for a given document and a given pairwise weight vector $\mathbf{w}$. Then, during inference, a clustering $\mathcal{C}$ is predicted by maximizing the scoring function $s(\mathcal{C}; \mathbf{w}, d)$, over all valid (i.e. satisfying symmetry and transitivity) clustering binary functions $\mathcal{C} : \{1, \ldots, m_d\} \times \{1, \ldots, m_d\} \rightarrow \{0, 1\}$.

3.1 Mention Pair Scorer

We model the task of coreference resolution using a pairwise scorer which indicates the compatibility of a pair of mentions. The inference routine then predicts the final clustering — a structured prediction problem — using these pairwise scores.

Specifically, for any two mentions i and j (w.l.o.g. $j < i$), we produce a pairwise compatibility score w_{ji} using extracted features $\phi(j, i)$ as

$$w_{ji} = \mathbf{w} \cdot \phi(j, i) \ , \qquad (1)$$

where $\mathbf{w}$ is a weight parameter that is learned.

3.2 Latent Left-Linking Model

Our inference algorithm is inspired by the best-left-link approach. In particular, the score $s(\mathcal{C}; d, \mathbf{w})$ is defined so that each mention links to the antecedent mention (to its left) with the highest score (as long as the score is above some threshold, say, 0). Specifically:

$$s(\mathcal{C}; d, \mathbf{w}) = \sum_{i=1}^{m_d} \max_{0 \leq j < i, \mathcal{C}(i,j)=1} \mathbf{w} \cdot \phi(j, i) \ . \qquad (2)$$

In order to simplify the notation, we introduce a dummy mention with index 0, which is to the left (i.e. appears before) of all other mentions and has $w_{0i} = 0$ for all *actual* mentions $i > 0$. For a given clustering $\mathcal{C}$, if a mention i is not co-clustered with any previous actual mention $j, 0 < j < i$, then we assume that i links to 0 and $\mathcal{C}(i, 0) = 1$. In other words, $C(i, 0) = 1$ *iff* i is the first actual item of a cluster in $\mathcal{C}$. However, such an item i is **not** considered to be co-clustered with 0 and for any valid clustering, item 0 is always in a singleton dummy cluster, which is eventually discarded. The important property of the score s is that it is exactly maximized

by the best-left-link inference, as it maximizes individual left link scores and the creation of one left-link does not affect the creation of other left-links.

3.3 Learning

We use a max-margin approach to learn $\mathbf{w}$. We are given a training set D of documents where for each document $d \in D$, $\mathcal{C}_d$ refers to the annotated ground truth clustering. Then we learn $\mathbf{w}$ by minimizing

$$L(\mathbf{w}) = \frac{\lambda}{2}\|\mathbf{w}\|^2 + \frac{1}{|D|}\sum_{d\in D}\frac{1}{m_d}\Big(\max_{\mathcal{C}}\big(s(\mathcal{C};d,\mathbf{w}) + \Delta(\mathcal{C},\mathcal{C}_d)\big) - s(\mathcal{C}_d;d,\mathbf{w})\Big)\ ,$$

where $\Delta(\mathcal{C},\mathcal{C}_d)$ is a loss function used in coreference. In order to achieve tractable loss-augmented minimization — something not possible with standard loss functions used in coreference (e.g. B^3 (Bagga and Baldwin, 1998)) — we use a decomposable loss function that just counts the number of mention pairs on which $\mathcal{C}$ and $\mathcal{C}_d$ disagree: $\Delta(\mathcal{C},\mathcal{C}_d) = \sum_{i,j=0,j<i}^{m_d}\mathbb{I}_{\mathcal{C}(i,j)=\mathcal{C}_d(i,j)}$, where $\mathbb{I}$ is a binary indicator function. This loss function is equivalent to the numerator of the Rand index loss (Rand, 1971). With this form of loss function and using the scoring function in Eq. (2), we can write $L(\mathbf{w})$ as

$$\frac{\lambda}{2}\|\mathbf{w}\|^2 + \frac{1}{|D|}\sum_{d\in D}\frac{1}{m_d}\sum_{i=1}^{m_d}\Big(\max_{0\le j<i}\big(\mathbf{w}\cdot\phi(j,i) + \delta(\mathcal{C}_d,i,j)\big) - \max_{0\le j<i, C(i,j)=1}(\mathbf{w}\cdot\phi(j,i))\Big)\ , \quad (3)$$

where $\delta(\mathcal{C}_d,i,j) = 1 - \mathcal{C}_d(i,j)$ is the loss-based margin that is 1 if i and j are not coreferent in $\mathcal{C}_d$, and is 0 otherwise. In the above objective function, the left-links remain latent while we get to observe the clustering. This objective function is related to latent structural SVMs (Yu and Joachims, 2009). However Yu and Joachims (2009) use a spanning tree based latent structure which does not have the left-to-right directionality we exploit. We can minimize the above function using Concave Convex Procedure (Yuille and Rangarajan, 2003), which is guaranteed to reach the local minima. However, such a procedure is costly as it requires doing inference on all the documents to compute a single gradient update. Consequently, we choose a faster stochastic sub-gradient descent (SGD) approach. Since $L(\mathbf{w})$ in Eq. (3) decomposes not only over training documents, but also over individual mentions in each document, we can perform SGD on a per-mention basis. The stochastic sub-gradient w.r.t. mention i in document d is given by

$$\begin{aligned}\nabla L(\mathbf{w})_d^i &\propto \phi(j',i) - \phi(j'',i) + \lambda\mathbf{w}, \text{ where} \quad (4)\\ j' &= \arg\max_{0\le j<i}(\mathbf{w}\cdot\phi(j,i) + 1 - \mathcal{C}_d(i,j))\\ j'' &= \arg\max_{0\le j<i, C(i,j)=1}\mathbf{w}\cdot\phi(j,i)\end{aligned}$$

While SGD has no theoretically convergence guarantee, it works excellently in our experiments. Specifically, we observe that SGD achieves similar training performance to CCCP with a speed-up of around 10,000.

3.4 Incorporating Constraints

Next, we show how to incorporate domain knowledge-based constraints into L^3M and generalize it to CL^3M. In CL^3M, we obtain a clustering by maximizing a constraint-augmented scoring function f given by

$$s(\mathcal{C};d,\mathbf{w}) + \sum_{p=1}^{n_c}\rho_p\psi_p(d,\mathcal{C}),$$

where the second term on the R.H.S. is the score contributed by domain specific constraints $\psi_1,\ldots,\psi_{n_c}$ with their respective scores $\rho_1,\ldots,\rho_{n_c}$. In particular, $\psi_p(d,\mathcal{C})$ measures the extent to which a given clustering $\mathcal{C}$ satisfies the p^{th} constraint. Note that this framework is general and can be applied to inject mention-to-cluster or cluster-to-cluster level constraints too. However, for simplicity, we consider here only constraints between mention pairs. This allows us derive fast greedy algorithm to solve the inference problem. The details of our constraints are presented in Sec. 5.

All of our constraints can be categorized into two groups: "must-link" and "cannot-link"."Must-link" constraints encourage a pair of mentions to connect, while "cannot-link" constraints discourage mention pairs from being linked. Consequently, the coefficients ρ_p associated with "must-link" constraints are positive while ρ_p for "cannot-link" constraints are negative. In the following, we briefly discuss how to

solve the inference problem with these two types of constraints.

We slightly abuse notations and use $\psi_p(j,i)$ to indicate the p^{th} constraint on a pair of mentions (i,j). $\psi_p(j,i)$ is a binary function that is 1 *iff* two mentions i and j satisfy the conditions specified in constraint p. Chang et al. (2011) shows that best-left-link inference can be formulated as an ILP problem. When we add constraints, the ILP becomes:

$$\begin{aligned} \underset{B,C\in\{0,1\}}{\arg\max} \quad & \sum_{i,j:j<i} w_{ji}B_{ji} + \sum_{i,j} \rho_p \psi_p(j,i) C_{ij} \\ \text{s.t} \quad & C_{kj} \geq C_{ij} + C_{ki} - 1,\ \forall i,j,k, \\ & \sum_{j=0}^{i-1} B_{ji} = 1,\ \forall i \\ & B_{ji} < C_{ji}, C_{ji} = C_{ji}, \forall i,j, \end{aligned} \tag{5}$$

where $C_{ij} \equiv C(i,j)$ is a binary variable indicating whether i and j are in the same cluster or not and B_{ji} is an auxiliary variable indicating the best-left-link for mention i. The first set of inequality constraints in (5) enforces the transitive closure of the clustering. The constraints $B_{ji} < C_{ji}, \forall i,j$ enforce the consistency between these two sets of variables.

One can use an off-the-shelf solver to solve Eq. (5). However, when the absolute values of the constraint scores ($|\rho_p|$) are high (the hard constraint case), then the following greedy algorithm approximately solves the inference efficiently. We scan the document from left-to-right (or in any other arbitrary order). When processing mention i, we find

$$j^* = \underset{j<i}{\arg\max} \quad w_{ji} + \sum_{k:\hat{C}(k,j)=1} \sum_p \rho_p \psi_p(k,i), \tag{6}$$

where $\hat{C}$ is the current clustering obtained from the previous inference steps. Then, we add a link between mention i and j^*. The rest of the inference process is the same as in the original best-left-link inference. Specifically, this inference procedure combines the classifier score for mention pair i,j, with the constraints score of all mentions currently co-clustered with j. We discuss this further in Section 5.

4 Probabilistic Latent Left-Linking Model

In this section, we extend and generalize our left-linking model approach to a probabilistic model, Probabilistic Latent Left-Linking Model (PL^{3}M), that allows us to naturally consider mention-to-entity (or mention-to-cluster) links. While in L^{3}M, we assumed that each mention links deterministically to the max-scoring mention on its left, in PL^{3}M, we assume that mention i links to mention j, $j \leq i$, with probability given by

$$Pr[j \leftarrow i; d, \mathbf{w}] = \frac{e^{\frac{1}{\gamma}(\mathbf{w}\cdot\phi(i,j))}}{Z_i(\mathbf{w},\gamma)} \ . \tag{7}$$

Here $Z_i(\mathbf{w},\gamma) = \sum_{0\leq k<i} e^{\frac{1}{\gamma}(\mathbf{w}\cdot\phi(i,k))}$ is a normalizing constant and $\gamma \in (0,1]$ is a constant temperature parameter that is tuned on a development set (Samdani et al., 2012). We assume that the event that mention i links to a mention j is independent of the event that mention i' links to j' for $i \neq i'$.

Inference with PL^{3}M: Given the probability of a link as in Eq. (7), the probability that mention i joins an existing cluster c, $Pr[c \odot i; d, \mathbf{w}]$, is simply the sum of the probabilities of i linking to the mentions inside c:

$$\begin{aligned} Pr[c \odot i; d, \mathbf{w}] &= \sum_{j\in c, 0\leq j<i} Pr[j \leftarrow i; d, \mathbf{w}] \\ &= \sum_{j\in c, 0\leq j<i} \frac{e^{\frac{1}{\gamma}(\mathbf{w}\cdot\phi(i,j))}}{Z_i(d,\mathbf{w},\gamma)} \ . \end{aligned} \tag{8}$$

Based on Eq. (8) and making use of the independence assumption of left-links, we follow a simple greedy clustering (or inference) algorithm: sequentially add each mention i to a previously formed cluster c^*, where $c^* = \arg\max_c Pr[c \odot i; d, \mathbf{w}]$. If the $\arg\max$ cluster is the singleton cluster with the dummy mention 0 (i.e. the score of all other clusters is below the threshold of 0), then i starts a new cluster and is not included in the dummy cluster. Note that we link a mention to a cluster taking into account all the mentions inside that cluster, mimicking the notion of a mention-to-cluster link. This provides more expressiveness than the Best-Left-Link inference, where a mention connects to a cluster solely based on a single pairwise link to some antecedent mention (the best-link mention) in that cluster.

The case of $\gamma = 0$: As γ approaches zero, it is easy to show that the probability $P[j \leftarrow i; d, w]$

in Eq. (7) approaches a Kronecker delta function that puts probability 1 on the *max-scoring mention* $j = \arg\max_{0 \leq k < i} \mathbf{w} \cdot \phi(i, j)$ (assuming no ties), and 0 everywhere else (Pletscher et al., 2010; Samdani et al., 2012). Consequently, as $\gamma \to 0$, $Pr[c \odot i; d, \mathbf{w}]$ in Eq. 8 approaches a Kronecker delta function centered on the cluster containing the max-scoring mention, thus reducing to the best-link case of L^3M. Thus, PL^3M, when tuning the value of γ, is a strictly more general model than L^3M.

Learning with PL^3M We use a likelihood-based approach to learning with PL^3M, and first compute the probability $Pr[\mathcal{C}; d, \mathbf{w}]$ of generating a clustering $\mathcal{C}$, given $\mathbf{w}$. We then learn $\mathbf{w}$ by minimizing the regularized negative log-likelihood of the data, augmenting the partition function with a loss-based margin (Gimpel and Smith, 2010). We omit the details of likelihood computation due to lack of space.

With PL^3M, we again follow a stochastic gradient descent technique instead of CCCP for the same reasons mentioned in Sec. 3.3. The stochastic gradient (subgradient when $\gamma = 0$) w.r.t. mention i in document d is given by

$$\nabla LL(\mathbf{w})^i_d \propto \sum_{0 \leq j < i} p_j \phi(i, j) - \sum_{0 \leq j < i} p'_j \phi(i, j) + \lambda \mathbf{w},$$

where p_j and p'_j, $j = 0, \ldots, i-1$, are non-negative weights that sum to one and are given by

$$p_j = \frac{e^{\frac{1}{\gamma}(\mathbf{w} \cdot \phi(i,j) + \delta(\mathcal{C}_d, i, j))}}{\sum_{0 \leq k < i} e^{\frac{1}{\gamma}(\mathbf{w} \cdot \phi(i,k) + \delta(\mathcal{C}_d, i, k))}} \text{ and}$$

$$p'_j = \frac{\mathcal{C}_d(i, j) Z_i(d, \mathbf{w}, \gamma)}{Z_i(\mathcal{C}_d; d, \mathbf{w}, \gamma)} Pr[j \leftarrow i; d, \mathbf{w}] \ .$$

Interestingly, the above update rule generalizes the one for L^3M, as we are incorporating a weighted sum of all previous mentions in the update rule. With $\gamma \to 0$, the SGD in Eq. (4) converges to the SGD update in L^3M (Eq. (4)). Finally, in the presence of constraints, we can fold them inside the pairwise link probabilities as in Eq. (6).

5 Experiments and Results

In this section, we present our experiments on the two commonly used benchmarks for coreference — Ontonotes-5.0 (Pradhan et al., 2012) and ACE 2004 (NIST, 2004). Table 1 exhibits our bottom line results: CL^3M achieves the best result reported on Ontonotes-5.0 development set and essentially ties with (Fernandes et al., 2012) on the test set. As shown in Table 3, CL^3M is also the best algorithm on ACE and when evaluated on the gold mentions of Ontonotes. We show that CL^3M performs particularly well on clusters containing named entity mentions, which are more important for many information extraction applications. In the rest of this section, after describing our experimental setting, we provide careful analysis of our algorithms and compare them to competitive coreference approaches in the literature.

5.1 Experimental Setup

Datasets: ACE 2004 contains 443 documents — we used a standard split of these documents into 268 training, 68 development, and 106 testing documents used by Culotta et al. (2007) and Bengtson and Roth (2008). OntoNotes-5.0 dataset, released for the CoNLL 2012 Shared Task (Pradhan et al., 2012), is by far the largest annotated corpus on coreference. It contains 3,145 annotated documents drawn from a wide variety of sources — newswire, bible, broadcast transcripts, magazine articles, and web blogs. We report results on both development set and test set. To test on the development set, we further split the training data into training and development sets.

Classifier details: For each of the pairwise approaches, we assume the pairwise score is given by $\mathbf{w} \cdot \phi(\cdot, \cdot) + t$ where ϕ are the features, $\mathbf{w}$ is the weight vector learned by the approach, and t is a threshold which we set to 0 during learning (as in Eq. (1)), but use a tuned value (tuned on a development set) during testing. For learning with L^3M, we do stochastic gradient descent with 5 passes over the data. Empirically, we observe that this is enough to generate a stable model. For PL^3M (Sec. 4), we tune the value of γ using the development set picking the best γ from $\{0.0, 0.2, \ldots, 1.0\}$. Recall that when $\gamma = 0$, PL^3M is the same as L^3M. We refer to L^3M and PL^3M with incorporating constraints during inference as CL^3M and CPL^3M (Sec. 3.4), respectively.

Metrics: We compare the systems using three popular metrics for coreference — MUC (Vilain et al., 1995), BCUB (Bagga and Baldwin, 1998), and

Entity-based CEAF (CEAF$_e$) (Luo, 2005). Following, the CoNLL shared tasks (Pradhan et al., 2012), we use the average F1 scores of these three metrics as the main metric of comparison.

Features: We build our system on the publicly available Illinois-Coref system[1] primarily because it contains a rich set of features presented in Bengtson and Roth (2008) and Chang et al. (2012a) (the latter adds features for pronominal anaphora resolution). We also compare with the Best-Left-Link approach described by Bengtson and Roth (2008).

Constraints: We consider the following constraints in CL^{3}M and CPL^{3}M.

- SameSpan: two mentions must be linked to each other if they share the same surface text span and the number of words in the text span is larger than a threshold (set as 5 in our implementation).
- SameDetNom: two mentions must be linked to each other if both mentions start with a determiner and the [0,1] wordnet-based similarity score between the mention head words is above a threshold (set to 0.8).
- SameProperName: two mentions must be linked if they are both proper names and the similarity score measured by a named entity-based similarity metric, Illinois NESim[2], are higher than a threshold (set to 0.8). For a person entity we add additional rules to extract the first name, last name and professional title as properties.
- ModifierMismatch: the constraint prevents two mentions to be linked if the head modifiers conflict. For example, the constraint prevents "northern Taiwan" from linking to "southern Taiwan". We gather a list of mutual exclusive modifiers from the training data.
- PropertyMismatch: the constraint prevents two mentions to be linked if their properties conflict. For example, it prevents male pronouns to link to female pronouns and "Mr. Clinton" to link to "Mrs. Clinton" by checking the gender property. The properties we consider are gender, number, professional title and the nationality.

[1]The system is available at `http://cogcomp.cs.illinois.edu/page/software_view/Coref/`

[2]`http://cogcomp.cs.illinois.edu/page/software_view/NESim`

	MUC	BCUB	CEAF$_e$	AVG
Dev Set				
Stanford	64.30	70.46	46.35	60.37
(Chang et al., 2012a)	65.75	70.25	45.30	60.43
(Martschat et al., 2012)	66.76	71.91	47.52	62.06
(Björkelund and Farkas,)	67.12	71.18	46.84	61.71
(Chen and Ng, 2012)	66.4	71.8	**48.8**	62.3
(Fernandes et al., 2012)	**69.46**	71.93	48.66	63.35
L^{3}M	67.88	71.88	47.16	62.30
CL^{3}M	69.20	**72.89**	48.67	**63.59**
Test Set				
Stanford	63.83	68.52	45.36	59.23
(Chang et al., 2012a)	66.38	69.34	44.81	60.18
(Martschat et al., 2012)	66.97	70.36	46.60	61.31
(Björkelund and Farkas,)	67.58	70.26	45.87	61.24
(Chen and Ng, 2012)	63.7	69.0	46.4	59.7
(Fernandes et al., 2012)	**70.51**	71.24	**48.37**	**63.37**
L^{3}M	68.31	70.81	46.73	61.95
CL^{3}M	69.64	**71.93**	48.32	63.30

Table 1: Performance on OntoNotes-5.0 with predicted mentions. We report the F1 scores (%) on various coreference metrics (MUC, BCUB, CEAF). The column AVG shows the average scores of the three. We observe that PL^{3}M and CPL^{3}M (see Sec. 4) yields the same performance as L^{3}M and CL^{3}M, respectively as the tuned γ for all the datasets turned out to be 0.

While the "must-link" constraints described in the paper can be treated as features, due to their high precision, treating them as hard constraints (set ρ to a high value) is a safe and direct way to inject human knowledge into the learning model. Moreover, our framework allows a constraint to use information from previous decisions (such as "cannot-link" constraints). Treating such constraints as features will complicate the learning model.

5.2 Performance of the End-to-End System

We compare our system with the top systems reported in the CoNLL shared task 2012 as well as with the Stanford's publicly released rule-based system (Lee et al., 2013; Lee et al., 2011), which won the CoNLL 2011 Shared Task (Pradhan et al., 2011). Note that all the systems use the same annotations (e.g., gender prediction, part-of-speech tags, name entity tags) provided by the shared task organizers.

However, each system implements its own mention detector and pipelines the identified mentions into the coreference clustering component. Moreover, different systems use a different set of features. In order to partially control for errors on mention detection and better evaluate the clustering component in our coreference system, we will also present results on correct (gold) mentions in the next section.

Table 1 shows the end-to-end results. On the development set, only the best performing system of Fernandes et al. (2012) is better than L^{3}M, but this difference disappears when we use our system with constraints, CL^{3}M. Although our system is much simple, it achieves the best B^3 score on the test set and is competitive with the best system participated in the CoNLL shared task 2012.

Performance on named entities: The coreference annotation in Ontonotes 5.0 includes various types of mentions. However, not all mention types are equally interesting. In particular, clusters which contain at least one proper name or a named entity mention are more important for information extraction tasks like Wikification (Mihalcea and Csomai, 2007; Ratinov et al., 2011), cross-document coreference resolution (Bagga and Baldwin, 1998), and entity linking and knowledge based population (Ji and Grishman, 2011).

Inspired by this, we compare our system to the best systems in the CoNLL shared task of 2011 (Stanford (Lee et al., 2011)) and 2012 (Fernandes (Fernandes et al., 2012)) on the following specific tasks on Ontonotes-5.0.

- **ENT-C**: Evaluate the system on clusters that contain at least one proper name mention. We generate the gold annotation and system outputs by using the gold and predicted name entity tag annotations provided by the CoNLL shard task 2012. That is, if a cluster does not include any name entity mention, then it will be removed from the final clustering.
- **PER-C**: As in the construction of ENT-C, but here we only consider clusters which contain at least one "Person (PER)" entity.
- **ORG-C**: As in the construction of Entity-C, but here we only consider clusters which contain at least one "Organization (ORG)" entity.

Typically, the clusters that get ignored in the above definitions contain only first and second person pronouns (which often happens in transcribed discourse.) Also note that all the systems are trained with the same name entity tags, provided by the shared task organizers, and we use the same name entity tags to construct the specific clustering. Also, in order to further ensure fairness, we do not tune our system to favor the evaluation of these specific types of clusters. We chose to do so because we only have access to the system output of Fernandes et al. (2012).

Task	Stanford	Fernandes	L^{3}M	CL^{3}M
ENT-C	44.06	47.05	46.63	**48.02**
PER-C	34.04	36.43	37.01	**37.57**
ORG-C	25.02	26.23	26.22	**27.01**

Table 2: Performance on named entities for OntoNotes-5.0 data. We compare our system to Fernandes (Fernandes et al., 2012) and Stanford (Lee et al., 2013) systems.

Table 2 shows the results. The performance of all systems degrades when considering only clusters that contain name entities, indicating that ENT-C is actually a harder task than the original coreference resolution problem. In particular, resolving ORG coreferent clusters is hard, because names of organizations are sometimes confused with person names, and they can be referred to using a range of pronouns (including "we" and "it"). Overall, CL^{3}M outperforms all the competing systems on the clusters that contain at least one specific type of entity by a margin larger than that for the overall coreference.

5.3 Analysis on Gold Mentions

To better understand the contribution of our joint learning and clustering model, we present experiments assuming that gold mentions are given. The definitions of gold mentions in ACE and Ontonotes are different because Ontonotes-5.0 excludes singleton clusters in the annotation. In addition, Ontonotes includes longer mentions; for example, it includes NP and appositives in the same mention. We compare with the publicly available Stanford (Lee et al., 2011) and IllinoisCoref (Chang et al., 2012a) systems; the system of Fernandes et al. (2012) is not publicly available. In addition, we also compare with the following two structured prediction baselines that use the same set of features as L^{3}M and PL^{3}M.

	MUC	BCUB	CEAF$_e$	AVG
ACE 2004 Gold Ment.				
All-Link-Red.	77.45	81.10	77.57	78.71
Spanning	73.31	79.25	74.66	75.74
IllinoisCoref	76.02	81.04	77.6	78.22
Stanford	75.04	80.45	76.75	77.41
(Stoyanov and Eisner, 2012)	**80.1**	81.8	-	-
L^{3}M	77.57	81.77	78.15	79.16
PL^{3}M	78.18	82.09	79.21	79.83
CL^{3}M	78.17	81.64	78.45	79.42
CPL^{3}M	78.29	**82.20**	**79.26**	**79.91**
Ontonotes 5.0 Gold Ment.				
All-Link-Red.	83.72	75.59	64.00	74.44
Spanning	83.64	74.83	61.07	73.18
IllinoisCoref	80.84	74.29	65.96	73.70
Stanford	82.26	76.82	61.69	73.59
L^{3}M	83.44	78.12	64.56	75.37
PL^{3}M	83.97	78.25	65.69	75.97
CL^{3}M	84.10	78.30	68.74	77.05
CPL^{3}M	**84.80**	**78.74**	**68.75**	**77.43**

Table 3: Performance on ACE 2004 and OntoNotes-5.0. All-Link-Red. is based on correlational clustering; Spanning is based on latent spanning forest based clustering (see Sec. 2). Our proposed approach is L^{3}M (Sec. 3) and PL^{3}M (sec. 4). CL^{3}M and CPL^{3}M are the version with incorporating constraints.

1. **All-Link-Red:** a reduced and faster alternative to the correlational clustering based approach (Finley and Joachims, 2005). We implemented this algorithm as an ILP and droped one of the three transitivity constraints for each triplet of mention variables. Following Pascal and Baldridge (2009) and Chang et al. (2011) we observe that this slightly improves the accuracy over a pure correlation clustering approach, in addition to speeding up inference.
2. **Spanning:** the latent spanning forest based approach presented by Yu and Joachims (2009). We use the publicly available implementation provided by the authors[3] for the ACE data; since their CCCP implementation is slow, we implemented our own stochastic gradient descent version to scale it to the much larger Ontonotes data.

[3]Available at http://www.cs.cornell.edu/ cnyu/latentssvm/

Table 3 lists the results. Although L^{3}M is simple and use only the features defined on pairwise mentions, it compares favorably with all recently published results. Moreover, the probabilistic generalization of L^{3}M, PL^{3}M, achieves even better performance. For example, L^{3}M with $\gamma = 0.2$ improves L^{3}M with $\gamma = 0$ by 0.7 points in ACE 2004. In particular, This shows that considering more than a one left-links is helpful. This is in contrast with the predicted mentions where $\gamma = 0$ performed best. We suspect that this is because noisy mentions can hurt the performance of PL^{3}M that takes into account not just the best scoring links, but also weaker links which are likely to be less reliable (more false positives). Also, as opposed to what is reported by Yu and Joachims (2009), the correlation clustering approach performs better than the spanning forest approach. We think that this is because we compare the systems on different metrics than they did and also because we use exact ILP inference for correlational clustering whereas Yu and Joachims (2009) used approximate greedy inference.

Both L^{3}M and PL^{3}M can be benefit from using constraints. However, The constraints improve only marginally on the ACE 2004 data because ACE uses shorter phrases as mentions. Consequently, constraints designed for leveraging information from long mention spans are less effective. Overall, the experiments show that L^{3}M and PL^{3}M perform well on modeling coreference clustering.

5.4 Ablation Study of Constrains

Finally, we study the value of individual constraints by adding one constraint at a time to the coreference system starting with the simple L^{3}M model. The system with all the constraints added is the CL^{3}M model introduced in Table 1. We then remove individual constraints from CL^{3}M to assess its contribution. Table 4 shows the results on the Ontonotes dataset with predicted mentions. Overall, it is shown that each one of the constraints has a contribution, and that using all the constraints improves the performance of the system by 1.29% in the AVG F1 score. In particular, most of this improvement (1.19%) is due to the must-link constraints (the first four constraints in the table). The must-link constraints are more useful for L^{3}M as L^{3}M achieves higher precision than recall (e.g., the precision and

	MUC	BCUB	$CEAF_e$	AVG
L^3M	67.88	71.88	47.16	62.30
+SameSpan	68.27	72.27	47.73	62.75
+SameDetNom	68.79	72.57	48.30	63.22
+SameProperName	69.11	72.81	48.56	63.49
+ModifierMismatch	69.11	72.81	48.58	63.50
+PropertyMismatch (i.e. CL^3M)	69.20	72.89	48.67	63.59
-SameSpan	68.91	72.66	48.36	63.31
-SameDetNom	68.62	72.51	48.06	63.06
-SameProperName	68.97	72.69	48.50	63.39
-ModifierMismatch	69.12	72.80	48.63	63.52
-PropertyMismatch	69.11	72.81	48.58	63.50

Table 4: Ablation study on constraints. We first show cumulative performance on OntoNotes-5.0 data with predicted mentions as constraints are added one at a time into the coreference system. Then we demonstrate the value of individual constraints by leaving out one constraint at each time.

recall of L^3M are 78.38% and 67.96%, respectively in B^3). As a result, the must-link constraints, which aim at improving the recall, do better when optimizing F1.

6 Conclusions

We presented a principled yet simple framework for coreference resolution. Furthermore, we showed that our model can be augmented in a straightforward way with knowledge based constraints, to improve performance. We also presented a probabilistic generalization of this model that can take into account entity-mention links by considering multiple possible coreference links. We proposed a fast stochastic gradient-based learning technique for our model. Our model, while operating at mention pair granularity, obtains state-of-the-art results on OntoNotes-5.0, and performs especially well on mention clusters containing named entities. We provided a detailed analysis of our experimental results.

Acknowledgments Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20155. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government.

References

A. Bagga and B. Baldwin. 1998. Algorithms for scoring coreference chains. In *In The First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference*.

M. Bansal and D. Klein. 2012. Coreference semantics from web features. In *Proceedings of ACL*, Jeju Island, South Korea, July.

N. Bansal, A. Blum, and S. Chawla. 2002. Correlation clustering. In *Proceedings of the 43rd Symposium on Foundations of Computer Science*.

E. Bengtson and D. Roth. 2008. Understanding the value of features for coreference resolution. In *EMNLP*, 10.

A. Björkelund and R. Farkas.

K.-W. Chang, R. Samdani, A. Rozovskaya, N. Rizzolo, M. Sammons, and D. Roth. 2011. Inference protocols for coreference resolution. In *CoNLL Shared Task*.

K.-W. Chang, R. Samdani, A. Rozovskaya, M. Sammons, and D. Roth. 2012a. Illinois-coref: The UI system in the CoNLL-2012 Shared Task. In *CoNLL Shared Task*.

M. Chang, L. Ratinov, and D. Roth. 2012b. Structured learning with constrained conditional models. *Machine Learning*, 88(3):399–431, 6.

C. Chen and V. Ng. 2012. Combining the best of two worlds: A hybrid approach to multilingual coreference resolution. In *Joint Conference on EMNLP and CoNLL - Shared Task*.

J. Clarke and M. Lapata. 2006. Constraint-based sentence compression: An integer programming approach. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 144–151, Sydney, Australia, July. ACL.

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

P. Denis and J. Baldridge. 2007. Joint determination of anaphoricity and coreference resolution using integer programming. In *Proceedings of the Annual Meeting of the North American Association of Computational Linguistics (NAACL)*.

P. Denis and J. Baldridge. 2008. Specialized models and ranking for coreference resolution. In *EMNLP*, pages 660–669.

G. Durrett, D. Hall, and D. Klein. 2013. Decentralized entity-level modeling for coreference resolution. In *Proceedings of ACL*, August.

E. R. Fernandes, C. N. dos Santos, and R. L. Milidiú. 2012. Latent structure perceptron with feature induction for unrestricted coreference resolution. In *Joint Conference on EMNLP and CoNLL - Shared Task*.

T. Finley and T. Joachims. 2005. Supervised clustering with support vector machines. In *Proceedings of the International Conference on Machine Learning (ICML)*.

K. Ganchev, J. Graça, J. Gillenwater, and B. Taskar. 2010. Posterior regularization for structured latent variable models. *Journal of Machine Learning Research*.

K. Gimpel and N. A. Smith. 2010. Softmax-margin CRFs: Training log-linear models with cost functions. In *NAACL*.

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

H. Ji and R. Grishman. 2011. Knowledge base population: successful approaches and challenges. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*.

T. Koo, A. M. Rush, M. Collins, T. Jaakkola, and D. Sontag. 2010. Dual decomposition for parsing with non-projective head automata. In *EMNLP*.

H. Lee, Y. Peirsman, A. Chang, N. Chambers, M. Surdeanu, and D. Jurafsky. 2011. Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Proceedings of the CoNLL-2011 Shared Task*.

H. Lee, A. Chang, Y. Peirsman, N. Chambers, M. Surdeanu, and D. Jurafsky. 2013. Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, 39(4).

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

S. Martschat, J. Cai, S. Broscheit, É. Mújdricza-Maydt, and M. Strube. 2012. A multigraph model for coreference resolution. In *Joint Conference on EMNLP and CoNLL - Shared Task*, July.

A. McCallum and B. Wellner. 2003. Toward conditional models of identity uncertainty with application to proper noun coreference. In *The Conference on Advances in Neural Information Processing Systems (NIPS)*.

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

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

Vincent Ng. 2005. Supervised ranking for pronoun resolution: Some recent improvements. In *AAAI*, pages 1081–1086.

NIST. 2004. The ACE evaluation plan.

D. Pascal and J. Baldridge. 2009. Global joint models for coreference resolution and named entity classification. In *Procesamiento del Lenguaje Natural*.

P. Pletscher, C. S. Ong, and J. M. Buhmann. 2010. Entropy and margin maximization for structured output learning. In *ECML PKDD*.

S. Pradhan, L. Ramshaw, M. Marcus, M. Palmer, R. Weischedel, and N. Xue. 2011. Conll-2011 shared task: Modeling unrestricted coreference in ontonotes. In *CoNLL*.

S. Pradhan, A. Moschitti, N. Xue, O. Uryupina, and Y. Zhang. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *CoNLL 2012*.

M. Sammons Y. Tu V. Vydiswaran Q. Do, D. Roth. 2009. Robust, light-weight approaches to compute lexical similarity. Technical report.

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

A. Rahman and V. Ng. 2011a. Coreference resolution with world knowledge. In *ACL*, pages 814–824.

A. Rahman and V. Ng. 2011b. Ensemble-based coreference resolution. In *IJCAI*.

A. Rahman and V. Ng. 2011c. Narrowing the modeling gap: a cluster-ranking approach to coreference resolution. *JAIR*.

W.M. Rand. 1971. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850.

L. Ratinov and D. Roth. 2012. Learning-based multi-sieve co-reference resolution with knowledge. In *EMNLP*.

L. Ratinov, D. Roth, D. Downey, and M. Anderson. 2011. Local and global algorithms for disambiguation to wikipedia. In *Proc. of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Dan Roth and Wen Tau Yih. 2004. A linear programming formulation for global inference in natural language tasks. In *Proceedings of CoNLL-04*, pages 1–8.

R. Samdani, M. Chang, and D. Roth. 2012. Unified expectation maximization. In *NAACL*.

W. M. Soon, H. T. Ng, and D. C. Y. Lim. 2001. A machine learning approach to coreference resolution of noun phrases. *Comput. Linguist.*

V. Stoyanov and J. Eisner. 2012. Easy-first coreference resolution. In *COLING*, pages 2519–2534.

V. Stoyanov, N. Gilbert, C. Cardie, and E. Riloff. 2009. Conundrums in noun phrase coreference resolution: making sense of the state-of-the-art. In *ACL*.

M. Vilain, J. Burger, J. Aberdeen, D. Connolly, and L. Hirschman. 1995. A model-theoretic coreference

scoring scheme. In *Proceedings of the 6th conference on Message understanding*.

C. Yu and T. Joachims. 2009. Learning structural svms with latent variables. In *Proceedings of the International Conference on Machine Learning (ICML)*.

A. L. Yuille and A. Rangarajan. 2003. The concave-convex procedure. *Neural Computation*, 15(4).