A Joint Model for Entity Analysis: Coreference, Typing, and Linking

Greg Durrett and **Dan Klein**
Computer Science Division
University of California, Berkeley
{gdurrett,klein}@cs.berkeley.edu

Abstract

We present a joint model of three core tasks in the entity analysis stack: coreference resolution (within-document clustering), named entity recognition (coarse semantic typing), and entity linking (matching to Wikipedia entities). Our model is formally a structured conditional random field. Unary factors encode local features from strong baselines for each task. We then add binary and ternary factors to capture cross-task interactions, such as the constraint that coreferent mentions have the same semantic type. On the ACE 2005 and OntoNotes datasets, we achieve state-of-the-art results for all three tasks. Moreover, joint modeling improves performance on each task over strong independent baselines.[1]

1 Introduction

How do we characterize the collection of entities present in a document? Two broad threads exist in the literature. The first is coreference resolution (Soon et al., 2001; Ng, 2010; Pradhan et al., 2011), which identifies clusters of mentions in a document referring to the same entity. This process gives us access to useful information about the referents of pronouns and nominal expressions, but because clusters are local to each document, it is often hard to situate document entities in a broader context. A separate line of work has considered the problem of entity linking or "Wikification" (Cucerzan, 2007; Milne and Witten, 2008; Ji and Grishman, 2011), where mentions are linked to entries in a given knowledge base. This is useful for grounding proper entities, but in the absence of coreference gives an incomplete picture of document content itself, in that nominal expressions and pronouns are left unresolved.

In this paper, we describe a joint model of coreference, entity linking, and semantic typing (named entity recognition) using a structured conditional random field. Variables in the model capture decisions about antecedence, semantic type, and entity links for each mention. Unary factors on these variables incorporate features that are commonly employed when solving each task in isolation. Binary and higher-order factors capture interactions between pairs of tasks. For entity linking and NER, factors capture a mapping between NER's semantic types and Wikipedia's semantics as described by infoboxes, categories, and article text. Coreference interacts with the other tasks in a more complex way, via factors that softly encourage consistency of semantic types and entity links across coreference arcs, similar to the method of Durrett et al. (2013). Figure 1 shows an example of the effects such factors can capture. The non-locality of coreference factors make exact inference intractable, but we find that belief propagation is a suitable approximation technique and performs well.

Our joint modeling of these three tasks is motivated by their heavy interdependencies, which have been noted in previous work (discussed more in Section 7). Entity linking has been employed for coreference resolution (Ponzetto and Strube, 2006; Rahman and Ng, 2011; Ratinov and Roth, 2012) and coreference for entity linking (Cheng and Roth, 2013) as part of pipelined systems. Past work has

[1] System available at http://nlp.cs.berkeley.edu

Transactions of the Association for Computational Linguistics, vol. 2, pp. 477–490, 2014. Action Editor: Jason Eisner.
Submitted 8/2014; Revised 10/2014; Published November 1, 2014.

Figure 1: Coreference can help resolve ambiguous cases of semantic types or entity links: propagating information across coreference arcs can inform us that, in this context, *Dell* is an organization and should therefore link to the article on `Dell` in Wikipedia.

shown that tighter integration of coreference and entity linking is promising (Hajishirzi et al., 2013; Zheng et al., 2013); we extend these approaches and model the entire process more holistically. Named entity recognition is improved by simple coreference (Finkel et al., 2005; Ratinov and Roth, 2009) and knowledge from Wikipedia (Kazama and Torisawa, 2007; Ratinov and Roth, 2009; Nothman et al., 2013; Sil and Yates, 2013). Joint models of coreference and NER have been proposed in Haghighi and Klein (2010) and Durrett et al. (2013), but in neither case was supervised data used for both tasks. Technically, our model is most closely related to that of Singh et al. (2013), who handle coreference, named entity recognition, and relation extraction.[2] Our system is novel in three ways: the choice of tasks to model jointly, the fact that we maintain uncertainty about all decisions throughout inference (rather than using a greedy approach), and the feature sets we deploy for cross-task interactions.

In designing a joint model, we would like to preserve the modularity, efficiency, and structural simplicity of pipelined approaches. Our model's feature-based structure permits improvement of features specific to a particular task or to a pair of tasks. By pruning variable domains with a coarse model and using approximate inference via belief propagation, we maintain efficiency and our model is only a factor of two slower than the union of the individual models. Finally, as a structured CRF, it is conceptually no more complex than its component models and its behavior can be understood using the same intuition.

We apply our model to two datasets, ACE 2005 and OntoNotes, with different mention standards and layers of annotation. In both settings, our joint model outperforms our independent baseline models. On ACE, we achieve state-of-the-art entity linking results, matching the performance of the system of Fahrni and Strube (2014). On OntoNotes, we match the performance of the best published coreference system (Björkelund and Kuhn, 2014) and outperform two strong NER systems (Ratinov and Roth, 2009; Passos et al., 2014).

2 Motivating Examples

We first present two examples to motivate our approach. Figure 1 shows an example of a case where coreference is beneficial for named entity recognition and entity linking. *The company* is clearly coreferent to *Dell* by virtue of the lack of other possible antecedents; this in turn indicates that *Dell* refers to the corporation rather than to Michael Dell. This effect can be captured for entity linking by a feature tying the lexical item *company* to the fact that COMPANY is in the Wikipedia infobox for `Dell`,[3] thereby helping the linker make the correct decision. This would also be important for recovering the fact that the mention *the company* links to `Dell`; however, in the version of the task we consider, a mention like *the company* actually links to the Wikipedia article for `Company`.[4]

Figure 2 shows a different example, one where the coreference is now ambiguous but entity linking is transparent. In this case, an NER system based on surface statistics alone would likely predict that *Freddie Mac* is a PERSON. However, the Wikipedia article for Freddie Mac is unambiguous, which allows us to fix this error. The pronoun *his* can then be correctly resolved.

These examples justify why these tasks should be handled jointly: there is no obvious pipeline order for a system designer who cares about the perfor-

[2] Our model could potentially be extended to handle relation extraction or mention detection, which has also been addressed in past joint modeling efforts (Daumé and Marcu, 2005; Li and Ji, 2014), but that is outside the scope of the current work.

[3] Monospaced fonts indicate titles of Wikipedia articles.

[4] This decision was largely driven by a need to match the ACE linking annotations provided by Bentivogli et al. (2010).

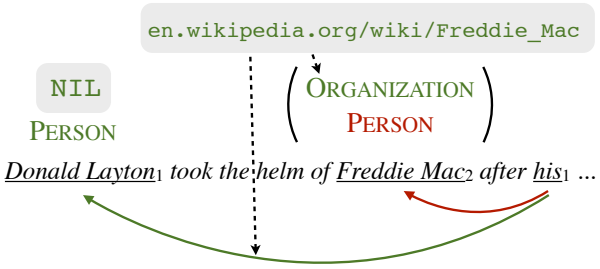

Figure 2: Entity links can help resolve ambiguous cases of coreference and entity types. Standard NER and coreference systems might fail to handle *Freddie Mac* correctly, but incorporating semantic information from Wikipedia makes this decision easier.

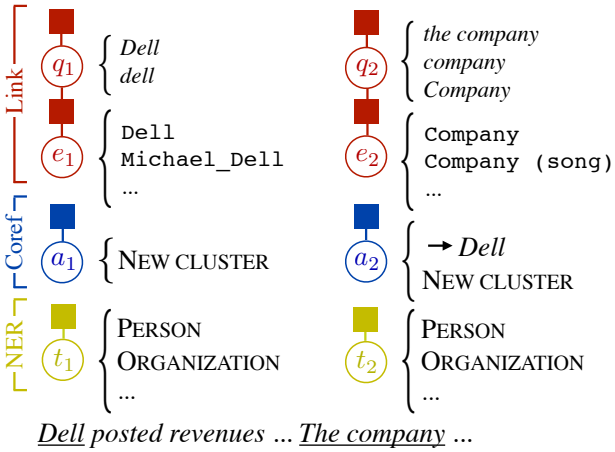

Figure 3: Random variables and task-specific factors present in our model. The a_i model coreference antecedents, the t_i model semantic types, the e_i model entity links, and the q_i are latent Wikipedia queries. Factors shown for each task integrate baseline features used when that task is handled in isolation. Coreference factors are described in Section 3.1.1, NER factors are described in Section 3.1.2, and entity linking factors are described in Section 3.1.3.

mance of the model on all three tasks.

3 Model

Our model is a structured conditional random field (Lafferty et al., 2001). The input (conditioning context) is the text of a document, automatic parses, and a set of pre-extracted mentions (spans of text). Mentions are allowed to overlap or nest: our model makes no structural assumptions here, and in fact we will show results on datasets with two different mention annotation standards (see Section 6.1 and Section 6.3).

Figure 3 shows the random variables in our model. We are trying to predict three distinct types of annotation, so we naturally have one variable per annotation type per mention (of which there are n):

- Coreference variables $\mathbf{a} = (a_1, \ldots, a_n)$ which indicate antecedents: $a_i \in \{1, \ldots, i-1, \text{NEW}\}$, indicating that the mention refers to some previous mention or that it begins a new cluster.
- Named entity type variables $\mathbf{t} = (t_1, \ldots, t_n)$ which take values in a fixed inventory of semantic types.[5]
- Entity link variables $\mathbf{e} = (e_1, \ldots, e_n)$ which take values in the set of all Wikipedia titles.

In addition we have variables $\mathbf{q} = (q_1, \ldots, q_n)$ which represent queries to Wikipedia. These are explained further in Section 3.1.3; for now, it suffices to remark that they are unobserved during both training and testing.

We place a log-linear probability distribution over these variables as follows:

$$p(\mathbf{a}, \mathbf{t}, \mathbf{e} | x; \theta) \propto \sum_{\mathbf{q}} \exp\left(\theta^\top f(\mathbf{a}, \mathbf{t}, \mathbf{e}, \mathbf{q}, x)\right)$$

where θ is a weight vector, f is a feature function, and x indicates the document text, automatic parses, and mention boundaries.

We represent the features in this model with standard factor graph notation; features over a particular set of output variables (and x) are associated with factors connected to those variables. Figure 3 shows the task-specific factors in the model, discussed next in Section 3.1. Higher-order factors coupling variables between tasks are discussed in Section 3.2.

3.1 Independent Model

Figure 3 shows a version of the model with only task-specific factors. Though this framework is structurally simple, it is nevertheless powerful enough for us to implement high-performing models for each task. State-of-the-art approaches to coreference (Durrett and Klein, 2013) and entity linking (Ratinov et al., 2011) already have this independent

[5] For the next few sections, we assume a fixed-mention version of the NER task, which looks like multi-way classification of semantic types. In Section 6.3.1, we adapt the model to the standard non-fixed-mention setting for OntoNotes.

structure and Ratinov and Roth (2009) note that it is a reasonable assumption to make for NER as well.[6] In this section, we describe the features present in the task-specific factors of each type (which also serve as our three separate baseline systems).

3.1.1 Coreference

Our modeling of the coreference output space (as antecedents chosen for each mention) follows the mention-ranking approach to coreference (Denis and Baldridge, 2008; Durrett and Klein, 2013). Our feature set is that of Durrett and Klein, targeting surface properties of mentions: for each mention, we examine the first word, head word, last word, context words, the mention's length, and whether the mention is nominal, proper or pronominal. Anaphoricity features examine each of these properties in turn; coreference features conjoin various properties between mention pairs and also use properties of the mention pair itself, such as the distance between the mentions and whether their heads match. Note that this baseline does not rely on having access to named entity chunks.

3.1.2 Named Entity Recognition

Our NER model places a distribution over possible semantic types for each mention, which corresponds to a fixed span of the input text. We define the features of a span to be the concatenation of standard NER surface features associated with each token in that chunk. We use surface token features similar to those from previous work (Zhang and Johnson, 2003; Ratinov and Roth, 2009; Passos et al., 2014): for tokens at offsets of $\{-2, -1, 0, 1, 2\}$ from the current token, we fire features based on 1) word identity, 2) POS tag, 3) word class (based on capitalization, presence of numbers, suffixes, etc.), 4) word shape (based on the pattern of uppercase and lowercase letters, digits, and punctuation), 5) Brown cluster prefixes of length 4, 6, 10, 20 using the clusters from Koo et al. (2008), and 6) common bigrams of word shape and word identity.

[6] Pairwise potentials in sequence-based NER are useful for producing coherent output (e.g. prohibiting configurations like O I-PER), but since we have so far defined the task as operating over fixed mentions, this structural constraint does not come into play for our system.

3.1.3 Entity Linking

Our entity linking system diverges more substantially from past work than the coreference or NER systems. Most entity linking systems operate in two distinct phases (Cucerzan, 2007; Milne and Witten, 2008; Dredze et al., 2010; Ratinov et al., 2011). First, in the candidate generation phase, a system generates a ranked set of possible candidates for a given mention by querying Wikipedia. The standard approach for doing so is to collect all hyperlinks in Wikipedia and associate each hyperlinked span of text (e.g. *Michael Jordan*) with a distribution over titles of Wikipedia articles it is observed to link to (`Michael_Jordan`, `Michael_I._Jordan`, etc.). Second, in the disambiguation phase, a learned model selects the correct candidate from the set of possibilities.

As noted by Hachey et al. (2013) and Guo et al. (2013), candidate generation is often overlooked and yet accounts for large gaps in performance between different systems. It is not always clear how to best turn the text of a mention into a query for our set of hyperlinks. For example, the phrase *Chief Executive Michael Dell* has never been hyperlinked on Wikipedia. If we query the substring *Michael Dell*, the highest-ranked title is correct; however, querying the substring *Dell* returns the article on the company.

Our model for entity linking therefore includes both predictions of final Wikipedia titles e_i as well as latent query variables q_i that model the choice of query. Given a mention, possible queries are all prefixes of the mention containing the head with optional truecasing or lemmatization applied. Unary factors on the q_i model the appropriateness of a query based on surface text of the mention, investigating the following properties: whether the mention is proper or nominal, whether the query employed truecasing or lemmatization, the query's length, the POS tag sequence within the query and the tag immediately preceding it, and whether the query is the longest query to yield a nonempty set of candidates for the mention. This part of the model can learn, for example, that queries based on lemmatized proper names are bad, whereas queries based on lemmatized common nouns are good.

Our set of candidates links for a mention is the set of all titles produced by some query. The bi-

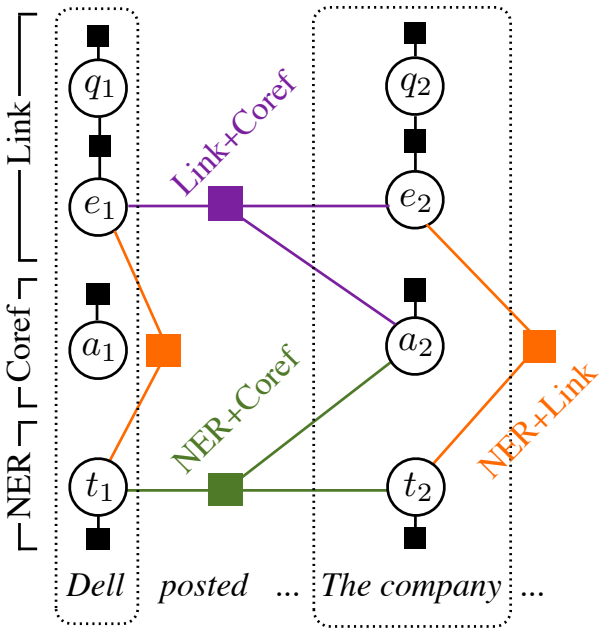

Figure 4: Factors that tie predictions between variables across tasks. Joint NER and entity linking factors (Section 3.2.1) tie semantic information from Wikipedia articles to semantic type predictions. Joint coreference and NER factors (Section 3.2.2) couple type decisions between mentions, encouraging consistent type assignments within an entity. Joint coreference and entity linking factors (Section 3.2.3) encourage relatedness between articles linked from coreferent mentions.

nary factors connecting q_i and e_i then decide which title a given query should yield. These include: the rank of the article title among all possible titles returned by that query (sorted by relative frequency count), whether the title is a close string match of the query, and whether the title matches the query up to a parenthetical (e.g. *Paul Allen* and `Paul_Allen_(editor)`).

We could also at this point add factors between pairs of variables (e_i, e_j) to capture coherence between choices of linked entities. Integration with the rest of the model, learning, and inference would remain unchanged. However, while such features have been employed in past entity linking systems (Ratinov et al., 2011; Hoffart et al., 2011), Ratinov et al. found them to be of limited utility, so we omit them from the present work.

3.2 Cross-task Interaction Factors

We now add factors that tie the predictions of multiple output variables in a feature-based way. Figure 4 shows the general structure of these factors. Each couples variables from one pair of tasks.

3.2.1 Entity Linking and NER

We want to exploit the semantic information in Wikipedia for better semantic typing of mentions. We also want to use semantic types to disambiguate tricky Wikipedia links. We use three sources of semantics from Wikipedia (Kazama and Torisawa, 2007; Nothman et al., 2013):

- Categories (e.g. `American financiers`); used by Ponzetto and Strube (2006; Kazama and Torisawa (2007; Ratinov and Roth (2012)
- Infobox type (e.g. `Person`, `Company`)
- Copula in the first sentence (*is a British politician*); used for coreference previously in Haghighi and Klein (2009)

We fire features that conjoin the information from the selected Wikipedia article with the selected NER type. Because these types of information from Wikipedia are of a moderate granularity, we should be able to learn a mapping between them and NER types and exploit Wikipedia as a soft gazetteer.

3.2.2 Coreference and NER

Coreference can improve NER by ensuring consistent semantic type predictions across coreferent mentions; likewise, NER can help coreference by encouraging the system to link up mentions of the same type. The factors we implement for these purposes closely resemble the factors employed for latent semantic clusters in Durrett et al. (2013). That structure is as follows:

$$\log F_{i-j}(a_i, t_i, t_j) = \begin{cases} 0 & \text{if } a_i \neq j \\ f(i, j, t_i, t_j) & \text{otherwise} \end{cases}$$

That is, the features between the type variables for mentions i and j does not come into play unless i and j are coreferent. Note that there are quadratically many such factors in the graph (before pruning; see Section 5), one for each ordered pair of mentions (j, i) with $j < i$. When scoring a particular configuration of variables, only a small subset of the factors is active, but during inference when we marginalize over all settings of variables, each of the factors comes into play for some configuration.

This model structure allows us to maintain uncertainty about coreference decisions but still propagate information along coreference arcs in a soft way.

Given this factor definition, we define features that should fire over *coreferent* pairs of entity types. Our features target:

- The pair of semantic types for the current and antecedent mention
- The semantic type of the current mention and the head of the antecedent mention, and the type of the antecedent and head of the current

We found such monolexical features to improve over just type pairs and while not suffering from the sparsity problems of bilexical features.

3.2.3 Coreference and Entity Linking

As we said in Section 2, coreferent mentions can actually have different entity links (e.g. `Dell` and `Company`), so encouraging equality alone is less effective for entity linking than it is for NER. Our factors have the same structure as those for coreference-NER, but features now target overall semantic relatedness of Wikipedia articles using the structure of Wikipedia by computing whether the articles have the same title, share any out links, or link to each other. More complex relatedness schemes such as those described in Ratinov et al. (2011) can be implemented in this framework. Nevertheless, these basic features still promise to help identify related articles as well as name variations by exploiting the abundance of entity mentions on Wikipedia.

4 Learning

Our training data consists of d documents, where a given document consists of a tuple $(x, C^*, \mathbf{t}^*, \mathbf{e}^*)$. Gold-standard labels for types ($\mathbf{t}^*$) and entity links ($\mathbf{e}^*$) are provided directly, while supervision for coreference is provided in the form of a clustering C^*. Regardless, we can simply marginalize over the uncertainty about $\mathbf{a}^*$ and form the conditional log-likelihood of the training labels as follows:

$$\mathcal{L}(\theta) = \sum_{i=1}^{d} \log \sum_{\mathbf{a}^* \in \mathcal{A}(C_i^*)} p(\mathbf{a}^*, \mathbf{t}_\mathbf{i}^*, \mathbf{e}_\mathbf{i}^* | x; \theta)$$

where $\mathcal{A}(C^*)$ is the set of antecedent structures consistent with the gold annotation: the first mention in a cluster must pick the NEW label and subsequent mentions must pick an antecedent from the set of those preceding them in the cluster. This marginalization over latent structure has been employed in prior work as well (Fernandes et al., 2012; Durrett and Klein, 2013).

We adapt this objective to exploit parameterized loss functions for each task by modifying the distribution as follows:

$$p'(\mathbf{a}, \mathbf{t}, \mathbf{e} | x; \theta) \propto p(\mathbf{a}, \mathbf{t}, \mathbf{e}, x) \exp\left[\alpha_c \ell_c(\mathbf{a}, C^*) + \alpha_t \ell_t(\mathbf{t}, \mathbf{t}^*) + \alpha_e \ell_e(\mathbf{e}, \mathbf{e}^*)\right]$$

where ℓ_c, ℓ_t, and ℓ_e are task-specific loss functions with weight parameters α. This technique, softmax-margin, allows us to shape the distribution learned by the model and encourage the model to move probability mass away from outputs that are bad according to our loss functions (Gimpel and Smith, 2010). As in Durrett and Klein (2013), we take $\alpha_c = 1$ and use ℓ_c as defined there, penalizing the model by $\alpha_{c,\text{FA}} = 0.1$ for linking up a mention that should have been nonanaphoric, by $\alpha_{c,\text{FN}} = 3$ for calling nonanaphoric a mention that should have an antecedent, and by $\alpha_{c,\text{WL}} = 1$ for picking the wrong antecedent for an anaphoric mention. ℓ_t and ℓ_e are simply Hamming distance, with $\alpha_t = 3$ and $\alpha_e = 0$ for all experiments. We found that the outcome of learning was not particularly sensitive to these parameters.[7]

We optimize our objective using AdaGrad (Duchi et al., 2011) with L_1 regularization and $\lambda = 0.001$. Our final objective is

$$\mathcal{L}(\theta) = \sum_{i=1}^{d} \log \sum_{\mathbf{a}^* \in \mathcal{A}(C_i^*)} p'(\mathbf{a}^*, \mathbf{t}_\mathbf{i}^*, \mathbf{e}_\mathbf{i}^* | x; \theta) + \lambda \|\theta\|_1$$

This objective is nonconvex, but in practice we have found that it is very stable. One reason is that for any mention that has fewer than two antecedents in its cluster, all elements of $\mathcal{A}(C^*)$ only contain one possibility for that mention, and even for mentions with ambiguity, the parameters that the model ends up learning tend to place almost all of the probability mass consistently on one antecedent.

[7] These parameters allow us to trade off contributions to the objective from the different tasks, addressing Singh et al. (2013)'s objection to single objectives for joint models.

	Dev						Test					
	MUC	B^3	CEAF_e	Avg.	NER	Link	MUC	B^3	CEAF_e	Avg.	NER	Link
INDEP.	77.95	74.81	71.84	74.87	83.04	73.07	81.03	74.89	72.56	76.16	82.35	74.71
JOINT	**79.41**	**75.56**	**73.34**	**76.10**	**85.94**	**75.69**	81.41	74.70	72.93	76.35	**85.60**	**76.78**
Δ	+1.46	+0.75	+1.50	+1.23	+2.90	+2.62	+0.42	−0.19	+0.37	+0.19	+3.25	+2.07

Table 1: Results on the ACE 2005 dev and test sets for the INDEP. (task-specific factors only) and JOINT models. Coreference metrics are computed using their reference implementations (Pradhan et al., 2014). We report accuracy on NER because the set of mentions is fixed and all mentions have named entity types. Coreference and NER are compared to prior work in a more standard setting in Section 6.3. Finally, we also report accuracy of our entity linker (including links to NIL); entity linking is analyzed more thoroughly in Table 2. Bolded values represent statistically significant improvements with $p < 0.05$ according to a bootstrap resampling test.

5 Inference

For both learning and decoding, inference consists of computing marginals for individual variables or for sets of variables adjacent to a factor. Exact inference is intractabile due to our factor graph's loopiness; however, we can still perform efficient inference using belief propagation, which has been successfully employed for a similar model (Durrett et al., 2013) as well as for other NLP tasks (Smith and Eisner, 2008; Burkett and Klein, 2012). Marginals typically converge in 3-5 iterations of belief propagation; we use 5 iterations for all experiments.

However, belief propagation would still be quite computationally expensive if run on the full factor graph as described in Section 3. In particular, the factors in Section 3.2.2 and Section 3.2.3 are costly to sum over due to their ternary structure and the fact that there are quadratically many of them in the number of mentions. The solution to this is to prune the domains of the coreference variables using a coarse model consisting of the coreference factors trained in isolation. Given marginals $p_0(a_i|x)$, we prune values a_i such that $\log p_0(a_i|x) < \log p_0(a_i^*|x) - k$ for a threshold parameter k, which we set to 5 for our experiments; this is sufficient to prune over 90% of possible coreference arcs while leaving at least one possible gold link for 98% of mentions.[8] With this optimization, our full joint model could be trained for 20 iterations on the ACE 2005 corpus in around an hour.

We use minimum Bayes risk (MBR) decoding, where we compute marginals for each variable under the full model and independently return the most likely setting of each variable. Note that for coreference, this implies that we produce the MBR antecedent structure rather than the MBR clustering; the latter is much more computationally difficult to find and would be largely the same, since the posterior distributions of the a_i are quite peaked.

[8] In addition to inferential benefits, pruning an arc allows us to prune entire joint coreference factors and avoid instantiating their associated features, which reduces the memory footprint and time needed to build a factor graph.

6 Experiments

We present results on two corpora. First, we use the ACE 2005 corpus (NIST, 2005): this corpus annotates mentions complete with coreference, semantic types (per mention), and entity links (also per mention) later added by Bentivogli et al. (2010). We evaluate on gold mentions in this setting for comparability with prior work on entity linking; we lift this restriction in Section 6.3.

Second, we evaluate on the OntoNotes 5 corpus (Hovy et al., 2006) as used in the CoNLL 2012 coreference shared task (Pradhan et al., 2012). This corpus does not contain gold-standard entity links, so we cannot evaluate this portion of our model, though the model still exploits the information from Wikipedia to make coreference and named entity decisions. We will compare to prior coreference and named entity work in the system mentions setting.

6.1 ACE Evaluation

We tokenize and sentence-split the ACE dataset using the tools bundled with Reconcile (Stoyanov et al., 2010) and parse it using the Berkeley Parser (Petrov et al., 2006). We use the train/test split from Stoyanov et al. (2009), Haghighi and Klein (2010), and Bansal and Klein (2012).

	Non-NILS			NILS			
	Prec.	Rec.	F_1	Prec.	Rec.	F_1	Accuracy
FAHRNI	81.15	78.10	79.60	41.25	61.10	49.25	76.87
INDEP.	80.26	76.30	78.23	33.39	54.47	41.40	74.71
JOINT	83.26	77.67	80.37	35.19	65.42	45.77	76.78
Δ over INDEP.	+3.00	+1.37	+2.14	+1.80	+10.95	+3.37	+2.07

Table 2: Detailed entity linking results on the ACE 2005 test set. We evaluate both our INDEP. (task-specific factors only) and JOINT models and compare to the results of the FAHRNI model, a state-of-the-art entity linking system. We compare overall accuracy as well as performance at predicting NILS (mentions not in the knowledge base) and non-NILS. The JOINT model roughly matches the performance of FAHRNI and gives strong gains over the INDEP. system.

Table 1 shows our results. Coreference results are reported using MUC (Vilain et al., 1995), B^3 (Bagga and Baldwin, 1998), and $CEAF_e$ (Luo, 2005), as well as their average, the CoNLL metric, all computed from the reference implementation of the CoNLL scorer (Pradhan et al., 2014). We see that the joint model improves all three tasks compared to the individual task models in the baseline.

More in-depth entity linking results are shown in Table 2. We both evaluate on overall accuracy (how many mentions are correctly linked) as well as two more specific criteria: precision/recall/F_1 of non-NIL[9] predictions, and precision/recall/F_1 of NIL predictions. This latter measure may be important if a system designer is trying to identify new entities in a document. We compare to the results of the best model from Fahrni and Strube (2014), which is a sophisticated discriminative model incorporating a latent model of mention scope.[10]

Our performance is similar to that of Fahrni and Strube (2014), though the results are not exactly comparable for two reasons. First, our models are trained on different datasets: Fahrni and Strube (2014) train on Wikipedia data whereas we train on the ACE training set. Second, they make use of the annotated head spans in ACE whereas we only use detected heads based on automatic parses. Note that this information is particularly beneficial for locating the right query because "heads" may be multi-word expressions such as *West Bank* as part of the phrase *southern West Bank*.

[9] NIL is a placeholder for mentions which do not link to an article in Wikipedia.

[10] On the TAC datasets, this FAHRNI model substantially outperforms Ratinov et al. (2011) and has comparable performance to Cheng and Roth (2013), hence it is quite competitive.

	Coref	NER	Link
INDEP.	74.87	83.04	73.07
INDEP+LINKNER		+1.85	+2.41
INDEP+COREFNER	+0.56	+1.15	
INDEP+COREFLINK	+0.48		−0.16
JOINT−LINKNER	+0.79	+1.28	−0.06
JOINT−COREFNER	+0.56	+1.94	+2.07
JOINT−COREFLINK	+0.85	+2.68	+2.57
JOINT	+1.23	+2.90	+2.62
JOINT/LATENTLINK	+1.26	+3.47	−18.8

Table 3: Results of model ablations on the ACE development set. We hold out each type of factor in turn from the JOINT model and add each in turn over the INDEP. model. We evaluate the coreference performance using the CoNLL metric, NER accuracy, and entity linking accuracy.

6.2 Model Ablations

To evaluate the importance of the different parts of the model, we perform a series of ablations on the model interaction factors. Table 3 shows the results of adding each interaction factor in turn to the baseline and removing each of the three interaction factors from the full joint model (see Figure 4).

Link–NER interactions. These joint factors are the strongest out of any considered here and give large improvements to entity linking and NER. Their utility is unsurprising: effectively, they give NER access to a gazetteer that it did not have in the baseline model. Moreover, our relatively rich featurization of the semantic information on Wikipedia allows the model to make effective use of it.

Coref–NER interactions. These are moderately beneficial to both coreference and NER. Having re-

liable semantic types allows the coreference system to be bolder about linking up mention pairs that do not exhibit direct head matches. Part of this is due to our use of monolexical features, which are fine-grained enough to avoid the problems with coarse semantic type matching (Durrett and Klein, 2013) but still effectively learnable.

Coref–Link interactions. These are the least useful of any of the major factors, providing only a small benefit to coreference. This is likely a result of the ACE entity linking annotation standard: a mention like *the company* is not linked to the specific company it refers to, but instead the Wikipedia article `Company`. Determining the relatedness of `Company` to an article like `Dell` is surprisingly difficult: many related articles share almost no outlinks and may not explicitly link to one another. Further feature engineering could likely improve the utility of these factors.

The last line of Table 3 shows the results of an experiment where the entity links were not observed during training, i.e. they were left latent. Unsurprisingly, the system is not good at entity linking; however, the model is still able to do as well or even slightly better on coreference and named entity recognition. A possible explanation for this is that even the wrong Wikipedia link can in many cases provide correct semantic information: for example, not knowing which *Donald Layton* is being referred to is irrelevant for the question of determining that he is a PERSON and may also have little impact on coreference performance. This result indicates that the joint modeling approach is not necessarily dependent on having all tasks annotated. The model can make use of cross-task information even when that information comes via latent variables.

6.3 OntoNotes Evaluation

The second part of our evaluation uses the datasets from the CoNLL 2012 Shared Task (Pradhan et al., 2012), specifically the coreference and NER annotations. All experiments use the standard automatic parses from the shared task and mentions detected according to the method of Durrett and Klein (2013).

Evaluating on OntoNotes carries with it a few complications. First, gold-standard entity linking annotations are not available; we can handle this by leaving the e_i variables in our model latent. Second, and more seriously, NER chunks are no longer the same as coreference mentions, so our assumption of fixed NER spans no longer holds.

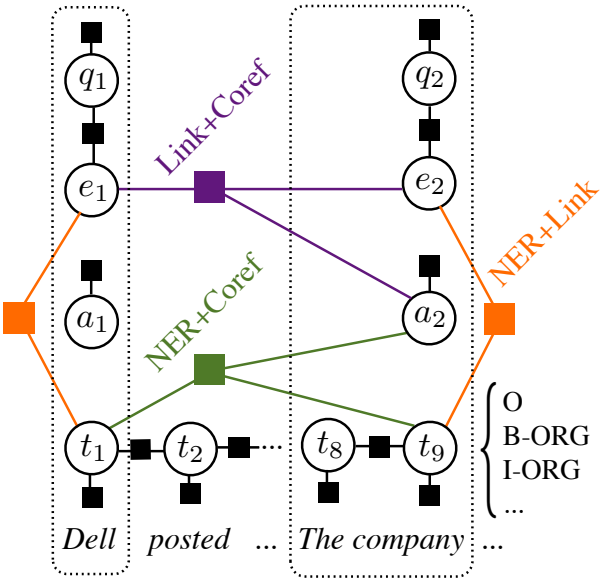

Figure 5: Modified factor graph for OntoNotes-style annotations, where NER chunks can now diverge from mentions for the other two tasks. NER is now modeled with token-synchronous random variables taking values in a BIO tagset. Factors coupling NER and the other tasks now interact with the NER chain via the NER nodes associated with the heads of mentions.

6.3.1 Divergent Coreference and NER

Our model can be adapted to handle NER chunks that diverge from mentions for the other two tasks, as shown in Figure 5. We have kept the coreference and entity linking portions of our model the same, now defined over system predicted mentions. However, we have replaced mention-synchronous type variables with standard token-synchronous BIO-valued variables. The unary NER features developed in Section 3.1.2 are now applied in the standard way, namely they are conjoined with the BIO labels at each token position. Binary factors between adjacent NER nodes enforce appropriate structural constraints and fire indicator features on transitions. In order to maintain tractability in the face of a larger number of variables and factors in the NER portion of our model, we prune the NER variables' domains using the NER model trained in isolation, similar to the procedure that we described for pruning coreference arcs in Section 5.

	MUC			B^3			CEAF_e			Avg.
	Prec.	Rec.	F_1	Prec.	Rec.	F_1	Prec.	Rec.	F_1	F_1
BERKELEY	72.85	65.87	69.18	63.55	52.47	57.48	54.31	54.36	54.34	60.33
FERNANDES	–	–	70.51	–	–	57.58	–	–	53.86	60.65
BJORKELUND	74.30	67.46	70.72	62.71	54.96	58.58	59.40	52.27	**55.61**	61.63
INDEP.	72.25	69.30	70.75	60.92	55.73	58.21	55.33	54.14	54.73	61.23
JOINT	72.61	69.91	**71.24**	61.18	56.43	**58.71**	56.17	54.23	55.18	**61.71**

Table 4: CoNLL metric scores for our systems on the CoNLL 2012 blind test set, compared to Durrett and Klein (2013) (the Berkeley system), Fernandes et al. (2012) (the winner of the CoNLL shared task), and Björkelund and Kuhn (2014) (the best reported results on the dataset to date). INDEP. and JOINT are the contributions of this work; JOINT improves substantially over INDEP. (these improvements are statistically significant with $p < 0.05$ according to a bootstrap resampling test) and achieves state-of-the-art results.

Cross-task factors that previously would have fired features based on the NE type for a whole mention now instead consult the NE type of that mention's head.[11] In Figure 5, this can be seen with factors involving e_2 and a_2 touching t_9 (*company*), the head of the second mention. Since the chain structure enforces consistency between adjacent labels, features that strongly prefer a particular label on one node of a mention will implicitly affect other nodes in that mention and beyond.

Training and inference proceed as before, with a slight modification: instead of computing the MBR setting of every variable in isolation, we instead compute the MBR sequence of labeled NER chunks to avoid the problem of producing inconsistent tag sequences, e.g. O I-PER or B-PER I-ORG.

6.3.2 Results

Table 4 shows coreference results from our INDEP. and JOINT models compared to three strong systems: Durrett and Klein (2013), Fernandes et al. (2012) (the winner of the CoNLL shared task), and Björkelund and Kuhn (2014) (the best reported results on the dataset). Our JOINT method outperforms all three as well as the INDEP. system.[12]

Next, we report results on named entity recognition. We use the same OntoNotes splits as for the coreference data; however, the New Testament (NT) portion of the CoNLL 2012 test set does not have gold-standard named entity annotations, so we omit it from our evaluation. This leaves us with exactly the CoNLL 2011 test set. We compare to two existing baselines from the literature: the Illinois NER system of Ratinov and Roth (2009) and the results of Passos et al. (2014). Table 5 shows that we outperform both prior systems in terms of F_1, though the ILLINOIS system features higher recall while our system features higher precision.

	Prec.	Rec.	F_1
ILLINOIS	82.00	**84.95**	83.45
PASSOS	–	–	82.24
INDEP.	83.79	81.53	82.64
JOINT	**85.22**	82.89	**84.04**
Δ over INDEP.	+1.43	+1.36	+1.40

Table 5: Results for NER tagging on the OntoNotes 5.0 / CoNLL 2011 test set. We compare our systems to the Illinois system (Ratinov and Roth, 2009) and the system of Passos et al. (2014). Our model outperforms both other systems in terms of F_1, and once again joint modeling gives substantial improvements over our baseline system.

7 Related Work

There are two closely related threads of prior work: those that address the tasks we consider in a different way and those that propose joint models for other related sets of tasks. In the first category, Hajishirzi et al. (2013) integrate entity linking into a sieve-based coreference system (Raghunathan et al., 2010), the aim being to propagate link decisions throughout coreference chains, block corefer-

[11] The NER-coreference portion of the model now resembles the skip-chain CRF from Finkel et al. (2005), though with soft coreference.

[12] The systems of Chang et al. (2013) and Webster and Curran (2014) perform similarly to the FERNANDES system; changes in the reference implementation of the metrics make exact comparison to printed numbers difficult.

ence links between different entities, and use semantic information to make additional coreference links. Zheng et al. (2013) build coreference clusters greedily left-to-right and maintain entity link information for each cluster, namely a list of possible targets in the knowledge base as well as a current best link target that is used to extract features (though that might not be the target that is chosen by the end of inference). Cheng and Roth (2013) use coreference as a preprocessing step for entity linking and then solve an ILP to determine the optimal entity link assignments for each mention based on surface properties of that mention, other mentions in its cluster, and other mentions that it is related to. Compared to these systems, our approach maintains greater uncertainty about all random variables throughout inference and uses features to capture cross-task interactions as opposed to rules or hard constraints, which can be less effective for incorporating semantic knowledge (Lee et al., 2011).

The joint model most closely related to ours is that of Singh et al. (2013), modeling coreference, named entity recognition, and relation extraction. Their techniques differ from ours in a few notable ways: they choose a different objective function than we do and also opt to freeze the values of certain variables during the belief propagation process rather than pruning with a coarse pass. Sil and Yates (2013) jointly model NER and entity linking in such a way that they maintain uncertainty over mention boundaries, allowing information from Wikipedia to inform segmentation choices. We could strengthen our model by integrating this capability; however, the primary cause of errors for mention detection on OntoNotes is parsing ambiguities rather than named entity ambiguities, so we would be unlikely to see improvements in the experiments presented here. Beyond maintaining uncertainty over mention boundaries, we might also consider maintaining uncertainty over the entire parse structure, as in Finkel and Manning (2009), who consider parsing and named entity recognition together with a PCFG.

8 Conclusion

We return to our initial motivation for joint modeling, namely that the three tasks we address have the potential to influence one another. Table 3 shows that failing to exploit any of the pairwise interactions between the tasks causes lower performance on at least one of them. Therefore, any pipelined system would necessarily underperform a joint model on whatever task came first in the pipeline, which is undesirable given the importance of these tasks. The trend towards broader and deeper NLP pipelines will only exacerbate this problem and make it more difficult to find a suitable pipeline ordering. In addition to showing that joint modeling is high-performing, we have also shown that it can be implemented with relatively low overhead, requiring no fundamentally new learning or inference techniques, and that it is extensible, due to its modular structure and natural partitioning of features. Taken together, these aspects make a compelling case that joint models can provide a way to integrate deeper levels of processing, particularly for semantic layers of annotation, and that this modeling power does not need to come at the expense of computational efficiency, structural simplicity, or modularity.

The Berkeley Entity Resolution System is available at `http://nlp.cs.berkeley.edu`.

Acknowledgments

This work was partially supported by BBN under DARPA contract HR0011-12-C-0014, by an NSF fellowship for the first author, and by a Google Faculty Research Award to the second author. Thanks to Angela Fahrni for helpful discussions about entity linking and for providing system output, to the anonymous reviewers for their insightful comments, and to our action editor Jason Eisner.

References

Amit Bagga and Breck Baldwin. 1998. Algorithms for Scoring Coreference Chains. In *Proceedings of the Conference on Language Resources and Evaluation Workshop on Linguistics Coreference*.

Mohit Bansal and Dan Klein. 2012. Coreference Semantics from Web Features. In *Proceedings of the Association for Computational Linguistics*.

Luisa Bentivogli, Pamela Forner, Claudio Giuliano, Alessandro Marchetti, Emanuele Pianta, and Kateryna Tymoshenko. 2010. Extending English ACE 2005 Corpus Annotation with Ground-truth Links to Wikipedia. In *Proceedings of the Workshop on*

The People's Web Meets NLP: Collaboratively Constructed Semantic Resources.

Anders Björkelund and Jonas Kuhn. 2014. Learning Structured Perceptrons for Coreference Resolution with Latent Antecedents and Non-local Features. In *Proceedings of the Association for Computational Linguistics*.

David Burkett and Dan Klein. 2012. Fast Inference in Phrase Extraction Models with Belief Propagation. In *Proceedings of the North American Chapter of the Association for Computational Linguistics*.

Kai-Wei Chang, Rajhans Samdani, and Dan Roth. 2013. A Constrained Latent Variable Model for Coreference Resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Xiao Cheng and Dan Roth. 2013. Relational Inference for Wikification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Silviu Cucerzan. 2007. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*.

Hal Daumé, III and Daniel Marcu. 2005. A Large-scale Exploration of Effective Global Features for a Joint Entity Detection and Tracking Model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Pascal Denis and Jason Baldridge. 2008. Specialized Models and Ranking for Coreference Resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Mark Dredze, Paul McNamee, Delip Rao, Adam Gerber, and Tim Finin. 2010. Entity Disambiguation for Knowledge Base Population. In *Proceedings of the International Conference on Computational Linguistics*.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, 12:2121–2159, July.

Greg Durrett and Dan Klein. 2013. Easy Victories and Uphill Battles in Coreference Resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, October.

Greg Durrett, David Hall, and Dan Klein. 2013. Decentralized Entity-Level Modeling for Coreference Resolution. In *Proceedings of the Association for Computational Linguistics*.

Angela Fahrni and Michael Strube. 2014. A Latent Variable Model for Discourse-aware Concept and Entity Disambiguation. In *Proceedings of the European Chapter of the Association for Computational Linguistics*.

Eraldo Rezende Fernandes, Cícero Nogueira dos Santos, and Ruy Luiz Milidiú. 2012. Latent Structure Perceptron with Feature Induction for Unrestricted Coreference Resolution. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Proceedings and Conference on Computational Natural Language Learning - Shared Task*.

Jenny Rose Finkel and Christopher D. Manning. 2009. Joint Parsing and Named Entity Recognition. In *Proceedings of the North American Chapter for the Association for Computational Linguistics*.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In *Proceedings of the Association for Computational Linguistics*.

Kevin Gimpel and Noah A. Smith. 2010. Softmax-Margin CRFs: Training Log-Linear Models with Cost Functions. In *Proceedings of the North American Chapter for the Association for Computational Linguistics*.

Yuhang Guo, Bing Qin, Yuqin Li, Ting Liu, and Sheng Li Li. 2013. Improving Candidate Generation for Entity Linking. In *Natural Language Processing and Information Systems.*

Ben Hachey, Will Radford, Joel Nothman, Matthew Honnibal, and James R. Curran. 2013. Evaluating Entity Linking with Wikipedia. *Artificial Intelligence*, 194:130–150, January.

Aria Haghighi and Dan Klein. 2009. Simple Coreference Resolution with Rich Syntactic and Semantic Features. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Aria Haghighi and Dan Klein. 2010. Coreference Resolution in a Modular, Entity-Centered Model. In *Proceedings of the North American Chapter of the Association for Computational Linguistics*.

Hannaneh Hajishirzi, Leila Zilles, Daniel S. Weld, and Luke Zettlemoyer. 2013. Joint Coreference Resolution and Named-Entity Linking with Multi-Pass Sieves. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust Disambiguation of Named Entities in Text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. OntoNotes: The 90% Solution. In *Proceedings of the North American Chapter of the Association for Computational Linguistics: Short Papers*.

Heng Ji and Ralph Grishman. 2011. Knowledge Base Population: Successful Approaches and Challenges. In *Proceedings of the Association for Computational Linguistics*.

Jun'ichi Kazama and Kentaro Torisawa. 2007. Exploiting Wikipedia as External Knowledge for Named Entity Recognition. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*.

Terry Koo, Xavier Carreras, and Michael Collins. 2008. Simple Semi-supervised Dependency Parsing. In *Proceedings of the Association for Computational Linguistics*.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the International Conference on Machine Learning*.

Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2011. Stanford's Multi-Pass Sieve Coreference Resolution System at the CoNLL-2011 Shared Task. In *Proceedings of the Conference on Computational Natural Language Learning: Shared Task*.

Qi Li and Heng Ji. 2014. Incremental Joint Extraction of Entity Mentions and Relations. In *Proceedings of the Association for Computational Linguistics*.

Xiaoqiang Luo. 2005. On Coreference Resolution Performance Metrics. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

David Milne and Ian H. Witten. 2008. Learning to Link with Wikipedia. In *Proceedings of the Conference on Information and Knowledge Management*.

Vincent Ng. 2010. Supervised Noun Phrase Coreference Research: The First Fifteen Years. In *Proceedings of the Association for Computational Linguistics*.

NIST. 2005. The ACE 2005 Evaluation Plan. In *NIST*.

Joel Nothman, Nicky Ringland, Will Radford, Tara Murphy, and James R. Curran. 2013. Learning Multilingual Named Entity Recognition from Wikipedia. *Artificial Intelligence*, 194:151–175, January.

Alexandre Passos, Vineet Kumar, and Andrew McCallum. 2014. Lexicon Infused Phrase Embeddings for Named Entity Resolution. In *Proceedings of the Conference on Computational Natural Language Learning*.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning Accurate, Compact, and Interpretable Tree Annotation. In *Proceedings of the Conference on Computational Linguistics and the Association for Computational Linguistics*.

Simone Paolo Ponzetto and Michael Strube. 2006. Exploiting Semantic Role Labeling, WordNet and Wikipedia for Coreference Resolution. In *Proceedings of the North American Chapter of the Association of Computational Linguistics*.

Sameer Pradhan, Lance Ramshaw, Mitchell Marcus, Martha Palmer, Ralph Weischedel, and Nianwen Xue. 2011. CoNLL-2011 Shared Task: Modeling Unrestricted Coreference in OntoNotes. In *Proceedings of the Conference on Computational Natural Language Learning: Shared Task*.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes. In *Joint Conference on EMNLP and CoNLL - Shared Task*.

Sameer Pradhan, Xiaoqiang Luo, Marta Recasens, Eduard Hovy, Vincent Ng, and Michael Strube. 2014. Scoring Coreference Partitions of Predicted Mentions: A Reference Implementation. In *Proceedings of the Association for Computational Linguistics*.

Karthik Raghunathan, Heeyoung Lee, Sudarshan Rangarajan, Nathanael Chambers, Mihai Surdeanu, Dan Jurafsky, and Christopher Manning. 2010. A Multi-Pass Sieve for Coreference Resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Altaf Rahman and Vincent Ng. 2011. Coreference Resolution with World Knowledge. In *Proceedings of the Association for Computational Linguistics: Human Language Technologies*.

Lev Ratinov and Dan Roth. 2009. Design Challenges and Misconceptions in Named Entity Recognition. In *Proceedings of the Conference on Computational Natural Language Learning*.

Lev Ratinov and Dan Roth. 2012. Learning-based Multi-sieve Co-reference Resolution with Knowledge. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*.

Lev Ratinov, Dan Roth, Doug Downey, and Mike Anderson. 2011. Local and Global Algorithms for Disambiguation to Wikipedia. In *Proceedings of the Association for Computational Linguistics*.

Avirup Sil and Alexander Yates. 2013. Re-ranking for Joint Named-Entity Recognition and Linking. In *Proceedings of the International Conference on Information and Knowledge Management*.

Sameer Singh, Sebastian Riedel, Brian Martin, Jiaping Zheng, and Andrew McCallum. 2013. Joint Inference of Entities, Relations, and Coreference. In *Proceedings of the 2013 Workshop on Automated Knowledge Base Construction*.

David A. Smith and Jason Eisner. 2008. Dependency Parsing by Belief Propagation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. 2001. A Machine Learning Approach to Coreference Resolution of Noun Phrases. *Computational Linguistics*, 27(4):521–544, December.

Veselin Stoyanov, Nathan Gilbert, Claire Cardie, and Ellen Riloff. 2009. Conundrums in Noun Phrase Coreference Resolution: Making Sense of the State-of-the-Art. In *Proceedings of the Association for Computational Linguistics*.

Veselin Stoyanov, Claire Cardie, Nathan Gilbert, Ellen Riloff, David Buttler, and David Hysom. 2010. Coreference Resolution with Reconcile. In *Proceedings of the Association for Computational Linguistics: Short Papers*.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A Model-Theoretic Coreference Scoring Scheme. In *Proceedings of the Conference on Message Understanding*.

Kellie Webster and James R. Curran. 2014. Limited Memory Incremental Coreference Resolution. In *Proceedings of the Conference on Computational Linguistics*.

Tong Zhang and David Johnson. 2003. A Robust Risk Minimization Based Named Entity Recognition System. In *Proceedings of the Conference on Natural Language Learning*.

Jiaping Zheng, Luke Vilnis, Sameer Singh, Jinho D. Choi, and Andrew McCallum. 2013. Dynamic Knowledge-Base Alignment for Coreference Resolution. In *Proceedings of the Conference on Computational Natural Language Learning*.